

(51) Int.Cl.<sup>6</sup>

G 11 B 27/00

識別記号

F I

G 11 B 27/00

D

D

審査請求 未請求 請求項の数4 O.L (全 65 頁)

(21)出願番号 特願平9-252000

(22)出願日 平成9年(1997)9月17日

(71)出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72)発明者 佐伯 慎一

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72)発明者 岡田 智之

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72)発明者 津賀 一宏

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(74)代理人 弁理士 滝木 智之 (外1名)

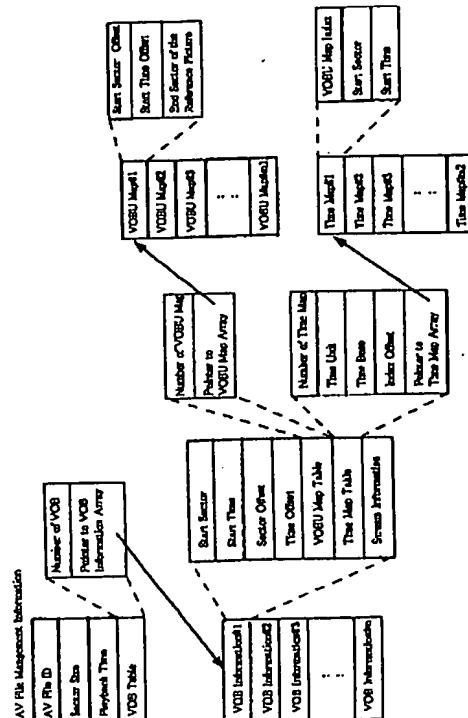
最終頁に続く

(54)【発明の名称】光ディスク及びその編集装置、再生装置

(57)【要約】

【課題】AVファイル管理情報の変更のための作業量  
や光ディスクへのアクセス回数を、AVファイルの大き  
さに依存しない一定値で押さえられる光ディスク及びそ  
の記録装置、再生装置の提供を目的とする。

【解決手段】1つ以上のファイルを管理するファイル  
システム管理領域と、AVデータを格納する1つ以上の  
AVファイルと、特殊再生情報ファイルとを有する光デ  
ィスクであって、特殊再生情報ファイルは、AVデータ  
内のVOBUのアドレスと再生時刻を示すVOBU情報とVOBU情  
報に記録された情報をAVファイルの先頭を基準にした  
値に補正するためのVOBU情報補正情報とを有することを  
特徴とする光ディスク。



## 【特許請求の範囲】

【請求項1】 1つ以上のファイルを管理するファイルシステム管理領域と、AVデータを格納する1つ以上のAVファイルと、特殊再生情報ファイルとを有する光ディスクであって、特殊再生情報ファイルは、AVデータ内のVOBUのアドレスと再生時刻を示すVOBU情報とVOBU情報に記録された情報をAVファイルの先頭を基準にした値に補正するためのVOBU情報補正情報とを有することを特徴とする光ディスク。

【請求項2】 1つ以上のファイルを管理するファイルシステム管理領域と、AVデータを格納する1つ以上のAVファイルと、特殊再生情報ファイルとを有する光ディスクであって、特殊再生情報ファイルは、基準時刻からの一定時間間隔の再生時刻に対応するVOBUを示すタイムマップ情報と、基準時刻を補正するためのタイムマップ基準情報を有することを特徴とする光ディスク。

【請求項3】 請求項1または請求項2記載の光ディスクの編集装置であって、編集によりAVファイルの先頭が変更されたときに、VOBU情報補正情報またはタイムマップ基準情報を生成する特殊再生情報補正手段を備えることを特徴とする編集装置。

【請求項4】 請求項3記載の編集装置により編集された光ディスクの再生装置であって、VOBU情報補正情報またはタイムマップ基準情報を参照してAVファイル内のVOBUのアドレスを求める手段を備えることを特徴とする再生装置。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、MPEGなどのAVデータを記録する光ディスク及びその編集装置、再生装置に関する。

## 【0002】

【従来の技術】 640MB程度が上限であった書き換え型光ディスクの分野で数GBの容量を有する相変化型光ディスクDVD-RAMが出現、コンピュータ用途だけでなくAVにおける記録・再生メディアとして期待されている。つまり従来の代表的なAV記録メディアである磁気テープに代わるメディアとして普及が予想される。

【0003】 (DVD-RAMの簡単な例) 近年、書き換え可能な光ディスクの高密度化が進みコンピュータデータやオーディオデータの記録に留まらず、動画像データの記録が可能となりつつある。

【0004】 例えば、光ディスクの信号記録面には、従来から凸凹状のガイド溝が形成されている。

【0005】 従来は凸または凹にのみ信号を記録していたが、ランド・グループ記録法により凸凹両方に信号を記録することが可能となった。これにより約2倍の記録密度向上が実現した(例えば特開平8-7282参考)。

【0006】 また、記録密度を向上させるために有効なCLV方式(線速度一定記録)の制御を簡易化し実用化を容易とするゾーンCLV方式なども考案、実用化されている(例えば特開平7-93873)。

【0007】 これらの大容量化を目指す光ディスクを用いて如何に動画像を含むAVデータを記録し、従来のAV機器を大きく超える性能や新たな機能を実現するかが今後の大きな課題である。

【0008】 このような大容量で書き換え可能な光ディスクの出現により、AVの記録・再生も従来の磁気テープに代わり光ディスクが主体となることが考えられる。テープからディスクへの記録メディアの移行はAV機器の機能・性能面でさまざまな影響を与えるものである。

【0009】 ディスクへの移行において最大の特徴はランダムアクセス性能の大幅な向上である。仮にテープをランダムアクセスする場合、一巻の巻き戻しに通常数分オーダーの時間が必要である。これは光ディスクメディアにおけるシーク時間(数10ms以下)に比べ桁違いに遅い。従ってテープは実用上ランダムアクセス装置になり得ない。

【0010】 具体例で補足を行うと、仮に2時間テープに2時間分の放送が一度記録された後、途中の30分だけを残し他の1時間半を消去し、消去した部分を他の録画に利用する場合を考える。テープ上には途中に残った30分の映像で分断された細切れの前半部分と後半部分が残る。理論上合計一時間半分の空き領域が存在するにも関わらず、一時間半分の連続録画を行うことはできない。

【0011】 これには次の2つの理由が存在する。まず第1にデジタルVTRであってもテープ上の映像データがファイルとして管理されていない。従ってどこが空き領域であり、どの部分が記録済み領域かが識別できない。第2の理由として、ランダムアクセス性能の低さにより連続領域以外の離れた領域にAVデータを連続記録も、再生もできない。

【0012】 このようにテープはその物理構造上、巻き戻しを行い先頭から連続的に記録するか、または記録された最後の部分に追記するかのいづれかの使用方法に限定されるものである。

【0013】 録画済みのテープに追加録画する場合、実質上先頭への巻き戻しが発生し記録済みの映像を消してしまったり、巻き戻し時間が掛かり録画チャンスを逃してしまうことをたびたび経験しているはずである。

【0014】 光ディスクによりもたらされるランダムアクセス性能の向上は単なる頭出しの早さだけではなく以下に説明する3つの機能的な特徴を生み出すものである。

1) AVデータがコンピュータのファイルとして取り扱える。

50 2) AVデータのノンリニア編集が可能となる。

3) 可変ビットレートのAVデータをリアルタイムで記録・再生可能である。

【0015】以下これらの特徴を従来技術により説明する。

(従来技術1:コンピュータファイルとしてAVデータを記録する)相変化型光ディスクDVD-RAMを搭載したマルチメディアPCの場合を例に取って従来技術1を説明する。

【0016】コンピュータに接続して光ディスクを用いる場合、OS中にあるファイルシステムと呼ばれる記録媒体管理プログラムを通してディスク上への書き込みや読み出しが行われる。コンピュータで主に扱うデータは文字やコードデータであるため1ファイルの容量は例えば100kb程度と光ディスクの記録容量(数GB)に対して小さい。そのためファイルシステムは多数の小さなファイルが記録・消去を繰り返しても問題が生じないように設計されている。

【0017】具体的にはディスク全体を数10KB単位の小さなデータブロックに分割し、一つのデータブロックにファイルがおさまらない場合には、必ずしも連続領域にない他のデータブロックへ残りのデータを記録する。最後にデータブロックのリンク情報をファイルの管理情報として記録しファイルの書き込みを完了する。何も記録されていないデータブロックは空き領域として管理され、必要に応じてデータの記録に使われる。またファイルが消去された場合には、使用済みのデータブロックは空き領域に登録される。

【0018】このような従来型のファイルシステム構造で光ディスクにAVデータを記録・再生する場合を考える。

【0019】まず、光ディスクに記録するAVデータのフォーマットはMPEG2(ISO/IEC13818参照)で規定され、ビットレートは8Mbpsであると仮定する。つまり1秒間に8メガビット(=1メガバイト)のデータを記録または読み出す必要がある。通常従来のマルチメディアPCではMPEG1と呼ばれる1.5Mbps程度のビットレートを有するAVデータが使用されるが、本例ではDVDにおいて標準的に使用されるMPEG2を基本に説明を行う。

【0020】ここで簡単に光ディスクドライブのデータ転送速度とランダムアクセス性能について説明する。

【0021】まずDVD-RAMにおける転送速度は最大11Mbps程度である。これは記録時であり読み出し時には倍速以上の読み出し速度の向上も可能である。ランダムアクセス性能はディスクドライブの機構系やサーボ系の設計により大きく変わる。通常データブロック間の移動に要する時間は、PC用の周辺機器において300ms程度であるが、民生用プレーヤなどでは最大1.5秒程度の時間を必要とする。特に最内周から最外周への移動に時間が掛かる。DVD-RAMへの記録を

PCで行った後、民生機で再生を行う場合を想定し、ここでは最大1.5秒が必要となると仮定する。

【0022】従ってファイルシステムの連続記録単位であるデータブロックを32KBとし、32KB毎にデータブロック間の移動が発生すると仮定すれば1秒分のデータである1MBを記録または読み出すのに、31回のブロック間移動が発生し最悪では4.5秒以上を要する。これではAVデータを途切れずに記録・再生することが全く保証できない。

【0023】何も記録されていないディスクにAVデータを記録する場合には連続するデータブロックを割り当てることができるため、連続記録・再生が可能である。しかし様々な記録と(部分)消去を何度も繰り返していくとデータブロックが連続して割り当てられなくなり同様の問題が生じる。

【0024】解決手段として従来はガーベッジコレクションと呼ばれる方法が存在する。これはデータブロックを連続的に配置できない場合に障害となるデータの配置を変更し、連続領域を確保する手法である。しかし光ディスクでは転送速度が容量の割に遅いため、ガーベッジコレクションに数時間を要することとなり実用的ではなかった。

【0025】(従来技術2:AVデータを編集する)ディスクによりAVデータを扱うことにより得られる第2の特徴は、AVデータのノンリニア編集が可能となる点である。これはAVデータ中の必要な部分を自由に書き換えたり、不要な部分だけを消去したり、複数のAVデータの任意の区間同士を接続したりというデータの処理性の向上とみなすことも出来る。

【0026】まず最初にテープを用いた編集について簡単に説明する。代表的なAVデータの処理として録画された映像のカット編集について考える。テープにおけるカット編集とは映像カット1と映像カット2を結合し最終映像を作成し、テープに記録する作業である。この作業をテープデッキで行う場合、最低2台のデッキと最低2本のテープ(素材と編集結果)が必要となる。

【0027】まずデッキ1に映像カット1の録画されたテープをセットし、デッキ2に編集結果を記録するテープをセットする。デッキ1で映像カット1の先頭個所まで頭だしを行った後デッキ1の再生とデッキ2の録画を同時に開始する。カット1の最後で2台のデッキを止める。続いてデッキ1のテープにある映像カット2の頭だしを行い、再度デッキ1の再生とデッキ2の録画を同時に開始する。カット2の最後まで録画が終われば、デッキ2のテープを巻き戻して編集作業が終了する。

【0028】テープメディアの編集がこのように手間と時間の掛かる作業であり且つ2台のデッキを必要とするため業務用やプロ用を除いて一般のコンシューマが編集を行うことは希である。

【0029】次にディスクを用いて編集を行う場合を考

える。テープにおける編集は映像をコピーすると考えられた。一方ディスクの場合にはカット1とカット2の間に接続性を指定するだけで編集作業が完了する。

【0030】つまり編集手順は以下の通りとなる。ここで上記の例と同様に映像カット1と映像カット2は同一のディスク上に記録されていると仮定する。

1) 映像カット1の最後と映像カット2の先頭を指定し接続を指示する。

2) (必要であれば)映像カット1と2以外の不要な部分を消去する。

【0031】再生時に映像カット1に続いてカット2が連続的に読み出されることにより、編集された結果が 출력される。

【0032】上記1)及び2)の操作は大量の映像データを読み出す必要がないため、極めて短時間の内に終了する。しかし従来光ディスクを用いてこのような編集をおこなうものは存在しなかった。

【0033】上記の編集において光ディスクを用いた場合の最大の課題は、光ディスクのランダムアクセス性能の低さである。テープに比べて桁違いに速い性能も編集操作には不十分である。そのため業務用のノンリニア編集機ではハードディスクが使用される。ハードディスクのランダムアクセス速度は数msであり、且つ転送速度も30M b p s以上であるため8M b p s程度のAVデータをリアルタイムで記録・再生することは容易である。

【0034】但し最大の問題は、ハードディスクは固定された装置であり、編集の入力媒体や最終出力媒体とはなり得ない点である。従って従来のノンリニア編集機の出力はあくまでビデオテープであり、将来光ディスクに記録する場合にも、ノンリニア編集機に内蔵される編集作業用ハードディスクと編集結果を書き出す光ディスクの2つのディスクが必要となる。

【0035】(従来技術3:可変ビットレートのAVデータをリアルタイムで記録・再生する)さらにランダムアクセス性能の向上により得られるディスクの第3のメリットとして可変ビットレートのAVデータが記録・再生可能となる。これはランダムアクセス性能の向上というより空回りできないテープメカと空回り可能なディスクメカの違いともいえる。

【0036】DVDではデータの読み出しに間欠転送と呼ぶ手法を用いている。つまりCDが絶えず連続的に一定の速度でデータを読み続けるのに対して、デコーダなど再生系でデータを消費する速度より速い転送レートでデータを読み出し、読み出し速度と消費速度の差により生じる余裕分を半導体メモリによりバッファリングを行う。バッファが一杯になった時点で読み出しを中断しキックバックと呼ばれる空回しを継続する。バッファに空きが生じると再度読み出しを行う。~

【0037】この手法により外部からの振動など何らかの原因でディスク読み出しにエラーが発生してもリトラ

イが可能となる。さらに、重要なことは記録されるAVデータのビットレートが可変にできる点である。

【0038】MPEG2は基本的に可変ビットレートでの圧縮・転送・記録を目指したものである。具体的にはエンコード時に入力されるビデオ信号の解析を行い画像の複雑度を計算する。複雑なシーンでは多くのビット量を割り当て、単純なシーンでは少ないビット量を割り当てる。動きの激しいシーンや高周波成分の多いシーンでは、静止しているシーンや低周波成分主体のシーンに比べ数倍のビット量が割り当てられる。

【0039】この技術により固定ビットレートのMPEGストリームに比べ最大2倍程度圧縮率を高め、長時間記録が可能となる。

【0040】しかし、可変ビットレートによりディスクにAVデータを記録すると、ランダムアクセス時にどこに飛び込めば良いかが分からぬという問題が生じる。

【0041】このためDVD-ROMでは、内部のビデオデータの再生時間が1つまたは複数のGOPで、0.4秒以上かつ1.0秒(例外的にVOBU終端では1.2秒)以下となる連続区間の先頭に、NVパックと呼ばれるDVD固有の情報を収めたパックをおき、NVパックに、近隣のNVパックを参照する情報や、最初のリファレンスピクチャを表示するために読み出さなければならぬデータの大きさが記録される。

【0042】なお、NVパックから次のNVパックの前までのAVデータを、VOBU(Video Object Unitの略)と呼ぶ。VOBUは、連続かつ隙間なくAVデータを分割する。

【0043】NVパックに記録される近隣のNVパックの参照のための情報として、そのVOBUの先頭のタイムコードを基準にして、ある時間間隔だけ離れた場所にある前後のVOBUのNVパックのアドレスを、自身のNVパックのアドレスからの相対値で示す。時間間隔は、1秒から1秒置きに1.5秒までと、2.0秒、6.0秒、12.0秒、24.0秒が使われる。

【0044】次に、早送り再生や巻戻し再生のような特殊再生の動作の一例について説明する。再生の速さに応じて、ある一定の時間間隔にあるVOBUのリファレンスピクチャのみを再生する事により、ほぼ一様な速度の特殊再生を実現する事ができる。一定の時間間隔のVOBUを次々と読み出すためには、NVパックのある近隣のNVパックのアドレスを示す情報を利用する。

【0045】また、AVデータの先頭から一定の時間間隔のタイムコードごとに、そのタイムコードに対応するVOBUのAVデータ内のアドレスを示すタイムマップ情報が記録される。タイムマップ情報を参照することにより、指定されたタイムコードからAVデータの再生を開始することができる。

【0046】【発明が解決しようとする課題】本発明は上記従来技術

において説明したディスクのランダムアクセス性能を最大限に引き出す上で支障となる以下の課題を解決し、書き換える可能な大容量光ディスクDVD-RAMの最大かつ本命の用途であるDVDレコーダを実現するものである。

【0047】従来技術で示したDVDビデオの光ディスクに記録されるタイムサーチマップやVOBUマップやNVパックのタイムコードのような特殊再生情報は、AVファイルの先頭を基準としたVOBUのアドレスやタイムコードを記録しているため、AVデータの編集によってAVファイルの先頭が削除されたり、AVファイルの先頭にデータが追加された場合には、記録されたすべてのアドレスやタイムコードを変更しなければならぬので、AVデータの大きさに比例する変更作業や光ディスクへのアクセスが発生するという問題があつた。

【0048】このため、本発明の目的の一つとして、AVファイル管理情報の変更のための作業量や光ディスクへのアクセス回数を、AVファイルの大きさに依存しない一定値で抑えられる光ディスク及びその編集装置、再生装置の提供を目的とする。

#### 【0049】

【課題を解決するための手段】上記課題を解決するためには、請求項1に係る発明においては、1つ以上のファイルを管理するファイルシステム管理領域と、AVデータを格納する1つ以上のAVファイルと、特殊再生情報ファイルとを有する光ディスクであつて、特殊再生情報ファイルは、AVデータ内のVOBUのアドレスと再生時刻を示すVOBU情報とVOBU情報に記録された情報をAVファイルの先頭を基準にした値に補正するためのVOBU情報補正情報を有することを特徴とする光ディスクとしている。

【0050】請求項2に係る発明においては、1つ以上のファイルを管理するファイルシステム管理領域と、AVデータを格納する1つ以上のAVファイルと、特殊再生情報ファイルとを有する光ディスクであつて、特殊再生情報ファイルは、基準時刻からの一定時間間隔の再生時刻に対応するVOBUを示すタイムマップ情報を、基準時刻を補正するためのタイムマップ基準情報を有することを特徴とする光ディスクとしている。

【0051】請求項3に係る発明においては、請求項1または請求項2記載の光ディスクの編集装置であつて、編集によりAVファイルの先頭が変更されたときに、VOBU情報補正情報を有するタイムマップ基準情報を生成する特殊再生情報補正手段を備えることを特徴とする編集装置としている。

【0052】請求項4に係る発明においては、請求項3記載の記録装置により記録された光ディスクの再生装置であつて、VOBU情報補正情報を有するタイムマップ基準情報を参照してAVファイル内でのVOBUのアドレスを求める手段を備えることを特徴とする再生装置としている。

#### 【0053】

【発明の実施の形態】本発明の1実施例であるDVDレコーダを用いて本発明の詳細を説明する。

#### 【0054】1. システム構成と各部の機能

DVDレコーダのシステム構成を図24を用いて説明する。

【0055】(ディスク記録部)ディスク記録部100は論理セクター番号と1つ以上のセクター単位の論理データ(2048バイト)を入力としてディスク上にデータを記録する。ディスク上に記録されるデータは16セクターの固まりであるECCブロック単位で処理されるため、ECCブロックの一部のセクタに記録するときには、以下の処理が発生する。

【0056】まず、記録したいセクターを含む当該ECCブロックを一度バッファメモリに読み出す。記録したいセクタ一分のデータをバッファの当該領域にコピーし、そのバッファのデータをECCブロックに記録する。

【0057】AVデータなど連続する大量のデータを記録する場合には、この処理のオーバーヘッドを取り除くため、ECCブロック単位で記録を行う。

【0058】(ディスク読み出し部)ディスク読み出し部101はディスク記録部と同様に読み出す論理セクタ番号とセクタ数を入力すると、当該ECCブロック単位で読み出しを行い、ECC処理を経て必要なセクタデータのみがファイルシステム部に転送される。ディスク記録部と同様にAVデータの読み出し時にECCブロック毎に16セクター単位で読み出しを行うことによりオーバーヘッドを削減する。

【0059】PC用の周辺機器であるDVD-RAMドライブはディスク記録部とディスク読み出し部の機能を実現し、SCSIやIDEと呼ばれるIFのコネクターでPC本体と接続される。PCをベースとするDVDレコーダは本説明と若干異なる仕様となる場合もあるが、本発明の趣旨とは無関係のため説明を省略する。

【0060】(ファイルシステム部)ファイルシステム部102はAVデータを扱うAVファイルシステム部103と制御情報など非AVファイルを扱う非AVファイルシステム部104で構成される。図1にファイルシステム部のコマンド一覧を示す。「CREATE」はディスク上にファイルを新しく作成し、ファイルディスクリプタを返す。「DELETE」はディスク上に存在するファイルを削除する。「OPEN」はディスク上に記録されているファイルにアクセスするために、そのファイルへのファイルディスクリプタを取得する。「CLOSE」はオープンされているファイルをクローズする。「WRITE」は非AVファイルをディスク上に記録する。「AV-WRITE」はAVファイルをディスク上に記録する。「READ」はディスク上に記録されたファイルを読み出す。「SEEK」はディスク上に記録されたデータストリーム内を移動する。「RENAME」はディスク上に記録されたファイルの名前を変更する。

「MKDIR」はディスク上に新しいディレクトリを作成する。「RMDIR」はディスク上に存在するディレクトリを削除する。「STATFS」はファイルシステムの現在の状況の問い合わせを行う。「GET\_ATTR」は現在オープンされているファイルの属性の問い合わせを行う。「SET\_ATTR」は現在オープンしているファイルの属性を変更する。「IN\_AV\_BLK\_BOUND」はAVファイルの指定された区間内にAVブロックの境界があるかどうかを調べる。「MERGE」はディスク上の2つAVファイルと、メモリ中のデータをマージする。「SPLIT」はディスク上のAVファイルを2つのAVファイルに分割する。「SHORTEN」はディスク上のAVファイルの端部を削除して、AVファイルの不必要的部分を削除する。「REPLACE」はAVファイルの一部分とメモリ中のデータを入れ替える。

【0061】ここで着目すべき点はAVデータの記録時には非AVデータの記録時とは異なるコマンドを用いる点である。これらの機能を行使した結果ディスク上の記録状態がどのように変化するかについては、後述するAVファイルシステムの動作例を参照のこと。

【0062】(ユーザIF部) ユーザIF部106は、DVレコーダ全体を制御している録画・編集・再生制御部105の指示により画面上にグラフィック表示を用いてユーザ操作を促したり、処理の途中経過を表示したり、ユーザのリモコン操作の結果を録画・編集・再生制御部に知らせたりする。

【0063】(録画・編集・再生制御部) 録画・編集・再生制御部105は、DVDレコーダ全体を制御する部分である。録画・編集・再生制御部105ではユーザ操作に応じて新規の録画、録画済みのAVデータの再生や編集をAVデータ録画部110、AVデータ再生部120、AVデータ編集部130に処理を要求する。この際、各処理部は独自にAVファイルシステム部103に対してAVデータの記録や読み出しを要求するが、非AVデータは制御データ管理部107があらかじめ主記憶上に読み出し、各処理部からの要求に応じて即座に情報を提供できる構成になっている。

【0064】(AVデータ録画部) AVデータ録画部110は、AVデータ入力部、AVファイル管理情報生成部、AVクリップ管理情報生成部、オーバーフロー対策部から構成される。AVデータ録画部110は、録画・編集・再生制御部105からの録画要求を受ける。

【0065】AVデータ入力部111は、入力される映像信号とオーディオ信号のMPEGデータへの変換、つまりエンコードをリアルタイムで実施する。次に、エンコードしたMPEGデータを、AVファイルとしてディスクに書き出すために、AVファイルシステム部103に渡す。また、AVデータ入力部111は、エンコードしてMPEGビデオデータのデコード可能な最小単位であるGOP毎に、そのGOPのデータをディスクに記録するときに必要なセクタ数とリファレンスピクチャのセクタ数を計

算し、得られた値をAVファイル管理情報生成部に渡す。

【0066】AVファイル管理情報生成部112は、GOPごとのセクタ数とリファレンスピクチャのセクタ数を録画が終了するまで主記憶上に記憶する。録画の終了時に、記憶していたGOP情報からAVファイル管理情報を生成し、制御データ管理部107に渡す。

【0067】AVクリップ管理情報生成部114は、録画開始時にAVデータの先頭にStartMarkをつけ、録画終了時にAVデータの終わりにEndMarkを付ける。この2つのマークを1つのクリップとして図38に示すAVクリップ管理情報を作成し、AVクリップ管理情報ファイルとして書き込む。また、このクリップ1つより構成される新しいクリップシークエンスをクリップシークエンスを管理するファイルに追加する。クリップとクリップシークエンスについては後で詳しく説明する。

【0068】次に、録画・編集・再生制御部から一時停止命令を受けた場合の処理を図25を用いて説明する。図25に示すAVデータ入力部は、ビデオエンコーダと、ビデオエンコーダの出力を格納するビデオバッファと、オーディオエンコーダと、オーディオエンコーダの出力を格納するオーディオバッファと、ビデオデータおよびオーディオデータを多重化するシステムエンコーダと、エンコーダの同期クロックであるSTC(システムタイムクロック)と、全体の制御および管理を行うエンコーダ制御部とから構成されている。エンコーダ制御部は、特にビデオエンコーダでのエンコードを管理している。具体的には、エンコードを行ったデータのGOP情報およびピクチャ情報などを管理し、AVファイル管理情報生成部に必要な情報を渡す。

【0069】録画・編集・再生制御部からの一時停止命令は、図25中のエンコーダ制御部に送られる。エンコーダ制御部は最初にビデオエンコーダおよびオーディオエンコーダに対して一時停止命令を出す。一時停止命令を受け取ったビデオエンコーダおよびオーディオエンコーダは、エンコーダ内部にバッファリングしているフレーム(入力中のフレームを含む)までのエンコードを行い、エンコードを中止する。ビデオエンコーダは、エンコードの中止と同時にエンコード制御部にエンコード中止を知らせる。ビデオエンコーダからエンコードの中止を受け取ったエンコード制御部は、システムエンコーダおよびSTC(システムタイムクロック)に一時停止命令を送り、システムエンコーダおよびSTC(システムタイムクロック)を一時停止させる。また、エンコーダ制御部は、AVクリップ管理情報生成部に対して、一時停止を行ったフレームのタイムコード(エンコード開始からの相対値)を送る。この後、録画・編集・再生制御部からの一時停止解除命令を受けることで、ビデオエンコーダ、オーディオエンコーダ、システムエンコーダおよびSTC(システムタイムクロック)は、エンコードを

再開する。

【0070】AVクリップ管理情報生成部では、一時停止時にAVデータ入力部より受け取ったタイムコードを一つのマークとしてAVクリップ管理情報に記録する。なお、一時停止時刻でクリップを分割しても良い。

【0071】オーバーフロー対策部113は、ディスクの記録速度に何らかの障害が生じ、あらかじめ想定されたビットレートでファイルシステム部に転送できない事態が発生した場合、AVデータ入力部で生成されるMPEGデータが内部に設けられたバッファメモリに対してオーバーフローを起こさないようにAVデータ入力部に対してビットレートの低下を指示するとともに、既に発生したMPEGデータから例えばB-pictureのデータのみを削除しビットレートを強制的に低下させる処理を行う。

【0072】(AVデータ編集部)AVデータ編集部120は、AVクリップ列編集部121、AVクリップ編集部122および特殊再生情報編集部123から構成される。

【0073】ここでAVクリップとAVクリップ列について説明する。AVクリップとはAVデータのカット編集において設定されるIN点(始まりの点)とOUT点(終わりの点)で指定されるAVデータの部分区間である。

【0074】図26はAVクリップとAVクリップ列の例を示す。上段はMPEGデータで構成されるAVファイル#jとAVファイル#kにおいて指定された3つのAVクリップの例を示す。クリップ#1はIN点をMark#1が示し、OUT点をMark#2が示す。Mark#1、Mark#2ともタイムコードで示される。クリップ#2、#3も同様にMarkによりIN点とOUT点が示される。このタイムコードはAVファイルの先頭を時刻00:00:00:00として計算したものである。

【0075】図26の下段は上段に示された3つのAVクリップがシーケンスとして列を構成したものである。まずAVファイル#jのクリップ#1、続いてAVファイル#kのクリップ#1、最後にAVファイル#jのクリップ#2が順番に並び編集結果として再生されるAVデータ列を表す。

【0076】ここで注意すべきはAVクリップ列編集部ではあくまでマークを指定するだけで、AVデータ自体にはなんの加工も施さない点である。このクリップ列情報はディスクに記録されるとともに、AVデータ再生部130において指定された順序で再生可能である。

【0077】AVクリップ編集部122は、AVクリップ列編集部121において指定されたAVクリップをAVファイルから切り出し、指定されたAVクリップ列の順番に接続をできるようにAVデータ自身を加工する。

【0078】ここでAVデータとAVクリップの関係を図27を用いて説明する。まずAVデータはMPEGで構成されるため、データは約0.5秒を一つの単位とするGOP(Group of Pictures)に分割されている。図中において#1から#8までの各VOBUはGOPを示す。VOBUはVideo Object

Unitの略であり、通常は1GOPで構成される。一方前述したようにAVクリップのIN点とOUT点を示すマークはタイムコードで指定されGOP(VOBU)の境界とは全く独立に指定される。

【0079】図において斜線部がAVクリップとAVデータ中の該当するビデオデータとの関係を概ね示している。しかしオーディオデータはビデオデータとはずれた状態で多重化されている。このようなMPEGデータに対してビデオを基準としたデータの取り出しを行うと、ビデオデータと同期の取れない(再生時刻がビデオよりも前の)オーディオデータが同時に取り出されてしまう。このようにAVクリップが指定されたからといってMPEGの性格上、AVデータを単純に切り出すことは困難である。このため切り出しにはMPEGデータの関係するGOP(VOBU)を再構成する必要が生じる。AVクリップ編集部ではこのようなMPEGデータの再構築を行う。詳細は動作例において説明する。

【0080】特殊再生情報編集部123は特殊再生情報生成部112において生成された特殊再生用ファイル内相対アドレス情報をAVクリップの編集にともなうAVデータの変形に応じて修正する。

【0081】(AVデータ再生部)AVデータ再生部は、AVクリップ列再生部132、AVファイル読み出し部133、読み出しエラー対策部134、AVデコーダ部131で構成される。

【0082】AVクリップ列再生部132は、録画・編集・再生制御部105から渡されたクリップシーケンスの再生を制御する。AVファイル読み出し部133に対して、クリップシーケンスを構成する各クリップのデータを読み出すことを要求する。AVデコーダ部131に対して、読み出されたデータをデコードすることを要求する。

【0083】AVファイル読み出し部133は、クリップシーケンスを構成する各クリップのAVファイル内のアドレス(ファイル・オフセット)を計算し、ファイルシステム部102に対して、AVファイルからのデータの読み出しを要求する。また、早送り再生や早戻し再生の場合には、再生のために最小限必要な部分を検索し、必要なデータのみを読み出すように、ファイルシステム部102に要求する。

【0084】読み出しエラー対策部134は、ディスクに記録されたAVデータがECC処理を実施してもなおエラーの訂正ができない場合、適切な回復、回避処置を行う。具体的には、次のGOP(VOBU)を代わりに指示したり、AVデコーダ部131のオーディオデコーダに対してオーディオのミュートを要求する。

【0085】AVデコーダ部131は、ファイルシステム部102から読み出されるAVデータのデコードを行う。デコーダのモデルは図28に示す。図28において150は入力されるAVデータ、151はAVデータ中

のパックヘッダに記述されたSCRやストリームIDに基づきデータをデコーダに転送するDeMUX（デマルチプレクサー）、152はビデオデコーダ、153はオーディオデコーダ、154はAVデータ中に存在する編集により生じたデータ境界における処理を行うシームレス接続処理部、155はビデオデコーダの出力である映像信号、156はオーディオデコーダの出力であるオーディオ信号である。本構成はシームレス接続処理部を除き通常のMPEGデコーダと同じである。

#### 【0086】2. ディスクフォーマット

次にディスクへ記録するフォーマットについて説明する。

【0087】まずディスク全体の構成を図2に示す。図において横軸は物理セクターアドレスを示す。物理セクターアドレスの先頭部分にはリードイン領域がありサポを安定させるために必要な基準信号や他のメディアとの識別信号などが記録されている。リードイン領域に続いてデータ領域が存在する。この部分に論理的に有効なデータが記録される。最後にリードアウト領域がありリードイン領域と同様な基準信号などが記録される。

【0088】以下本発明の対象となるデータ領域について詳述する。

（データ領域）まずデータ領域はセクタと呼ばれるアクセス可能な最小サイズのブロックに分割されて、セクタ毎に使用状況を割り当て情報管理領域を用いて管理される。割り当て管理情報領域の形態はリスト構造やテーブルが考えられる。ここではセクタビットマップという表を用いて管理する例を用いて説明する。図10にその様子を示す。

【0089】1つのセクタのサイズは2KBであり、ディスクへのREAD、WRITEはセクタサイズの整数倍の単位でしか許されない。更にデータ領域は複数のゾーンに分割される。本例では具体的なゾーン数は24とする。ゾーン導入の意味はゾーン内ではCAV（角速度一定、つまり回転速度一定）でディスクを回転させることにより記録時の制御を容易にすることにある。また各ゾーンには同一数のトラックが割り当てられる。本例ではゾーンあたり1888本とする。ゾーンの境界では各ゾーンで2トラック分(48から80セクターに相当)のバッファセクターが設けられ、この領域にはデータは記録できない。さらに各ゾーンの先頭はECCブロックの先頭セクターとなっており、アクセス性の向上を図っている。

【0090】このようにゾーン境界で記録ができないセクターが存在するが、使用上は不都合であるため物理セクターアドレスから表により論理セクターアドレスを計算することにより、論理セクターアドレスはデータの記録可能な領域のみを連続的に示すように考慮されている。従って以降はゾーンの境界に存在する領域は無視して議論を進めるが、但しゾーン境界を跨いで連続的にデータ記録や読み出しを行う場合には数100ms程度の遅延が発

生するものとする。

#### 【0091】3. AVファイルシステム

次にデータ領域の利用方法であるファイルシステムについて説明する。ファイルシステムの目的はアプリケーションがファイルを単位とするアクセスを可能とすること、つまり他のファイルのことは気にせずにアプリケーションを実現できることである。これにより多数のファイルが同一ディスク上に存在しても容易にアプリケーションが実現可能となる。さらにファイルの追記や消去を繰り返し行ってもデータ領域を有効に活用が出来る。これはファイルの内容となる論理データをディスク上に記録する際に小さなデータブロックに分割し、ブロック間のリンク情報を合わせて管理するからである。データ領域の先頭部分には、このようにファイルシステムを通してデータを管理するためのボリューム情報やファイル情報が記録される。これら管理情報に続いて実際のファイルデータが記録される。

【0092】ファイルデータにはAVデータと非AVデータの2種類が存在し、ディスク上における記録方法も異なる。これはAVデータの記録や読み出しにおけるリアルタイム性を保証すると同時に非AVデータ（通常小さい容量である）のファイルが多数記録された場合におけるディスクの使用効率を維持するためである。

【0093】図3はデータ領域の詳細である。リアルタイム性を保証するためにはデータの記録・読み出し時に発生するオーバーヘッド時間を正確に見積もる必要がある。まずデータ領域は前述のように24のゾーンに分かれ境界を跨いで連続記録・読み出しを行う際には数100msの遅延が生じる。一つのゾーンは固定長の論理ブロック（以後AVブロックと呼ぶ）に分割される。但しゾーンの最後のAVブロックは他のAVブロックよりも大きなサイズを有する。各AVブロックはECCブロックの整数倍で構成され、ECCブロックは16セクターで構成される。各セクターは2048Bのデータを記録可能である。図3は各セクターにAVデータを記録した場合の例である。AVブロックのサイズの決定方法に関しては後で説明する。

【0094】（AVブロック）各AVブロックはAVデータを記録しても良いし、非AVデータを記録しても良い。但し、AVデータと非AVデータを同一のAVブロック内に混在して記録することは認めない。AVブロックもセクタ同様、割り当て情報管理領域を用いて管理される。この形態もセクタと同様にリスト構造やテーブルが考えられる。

【0095】なお、この管理情報領域は、ディスクのボリューム管理領域の様にファイルシステム用の領域に持つことも出来るし、アプリケーションから処理できる1つのファイルとしてディスク上に持つことも出来る。

【0096】ここではテーブル構造である図4示すAVブロック管理テーブルを例に説明を行う。この管理テ

ブルはデータ領域の先頭にあるボリューム情報の一部として記録される。

【0097】管理テーブルはAVブロックアドレスに対応するAVブロックがAV用にデータが割り当てられているか、非AV用に割り当てられているか、未使用であるかを識別するために用いられる。図中に示すようにテーブルの内容である2ビットが00は未使用、01はAV、10は非AVを示す。各AVブロックの長さは本例では224ECCブロック（約7MB）であり、各ゾーンの最終AVブロックのみはテーブルに示すゾーン毎に異なる値となる。これはゾーン境界をAVブロックが跨ぐのを回避しつつ、ディスクの使用効率を維持するものである。

【0098】ここでAVブロックの長さはどのように決定されるのかを説明する。図5と図6はAVデータ記録時と読み出し時のバッファリングモデルを示すものである。図5においてエンコーダの出力であるMPEGデータはトラックバッファと呼ばれるFIFO(First In First Out)メモリに一時蓄積された後、ディスクに記録される。ディスクに記録する際には可能な限り連続したECCブロック単位でデータを記録することにより無駄な回転待ちやシーク時間を回避できる。AVブロックはこのように一度に連続的に記録・読み出しを行う単位である。この際にトラックバッファの占有状態をグラフ化した図が図5と図6である。

【0099】図5においてトラックバッファへの最大入力レートをVin、ディスク記録の許容最大レートをVoutとする。ここでVin < Voutである。本例では実際の値としてVin=8Mbps、Vout=11Mbpsと設定する。トラックバッファは初期値としてバッファ容量以下のある値までAVデータが蓄積済みと仮定する。これは蓄積せずに記録を開始するとVout > Vinのためすぐにトラックバッファがアンダーフローを起こし、最大許容レートでディスクに連続記録ができないためである。図5においてAVブロック#jの記録を行いながら、エンコードを続けるとディスクへの書き込み時間中はVin - Voutのレートでバッファの占有量は減少する。ブロック#jの記録が完了した時点で次のAVブロックであるAVブロック#kにジャンプ（シークと回転待ち）を行う。この間はエンコードのレートであるVinでバッファ占有量が増加する。ここで満足すべき条件はトラックバッファがオーバーフローし、AVデータが紛失することを避けることである。この条件はAVブロックの長さには無関係にジャンプ時間の最大値とVinから、

$Vin \times \text{最大ジャンプ時間} < \text{トラックバッファ容量}$ となる。

【0100】今、Vin=8Mbps、最大ジャンプ時間=1.5秒であるのでトラックバッファの容量は1.5MB必要である。もちろん記録時の最大ジャンプ時間が短いディスク記録装置を用いればバッファ容量は大幅に削減可能となる。

【0101】次に読み出し時のバッファリングモデルを図6を用いて説明する。図においてディスクから読み出されたデータはECC処理を経てトラックバッファへVinのレートで入力される。トラックバッファは記録時と同様のFIFOである。ワーストケースとしてトラックバッファに全くデータが溜まっていない状態からAVブロック#jの読み出しに入る。この場合読み出し中にはVin - Voutのレートでバッファにデータが溜まって行く。AVブロックの長さを224ECCブロック (=7.2MB)、Vin=11Mbps、Vout=8Mbpsとすると、AVブロックの読み込み時間は約5.2秒となりバッファをオーバーフローさせないためには約2MBのトラックバッファが必要となる。但し、トラックバッファがオーバーフローを起こしてもディスクに記録されているデータを紛失する訳ではないため、2MBのバッファを持つ必要はない。必要条件としては、AVブロック#jの読み込みを完了しジャンプしている間にトラックバッファがアンダーフローを起こさないだけのバッファ容量を確保することである。これは最大ジャンプ時間を1.5秒、Vout=8Mbpsとすると1.5MBとなる。最大ジャンプ時間が短い読み出し装置の場合、必要なバッファ容量は削減可能である。

【0102】ここでAVブロック長の下限値について説明する。上記の読み出し時のバッファリングモデルにおいてAVブロックを全て一気に読み出した場合2MBのトラックバッファを必要とするが、実際ジャンプ時にデータをアンダーフローさせないためには1.5MBのトラックバッファで十分であると説明した。つまり、AVブロック長を適切に設定したため、2MB > 1.5MBとなりAVブロック長を意識することなく議論を進めることができた。逆に一気に読み込んだ場合でも1.5MBに満たない量しかトラックバッファに蓄積できなかった場合を考えるとトラックバッファは明らかにアンダーフローを起こす。従ってAVブロック内のデータを読み出している間にトラックバッファに蓄積されるデータを、Vinのビットレートで読み出した場合に1.5秒以上の時間が必要になればよいから、

$$\text{AVブロック長 (bit単位)} \times (Vin - Vout) / (Vin \times Vin) > 1.5\text{秒}$$

つまり、AVブロック長は最低5.5MB以上必要となる。7MB程度に設定した理由はディスクエラーが発生した場合などのマージンを見込んでいるからである。

【0103】このAVブロックを用いてAVファイルを記録する場合、AVファイルの先頭と末尾部分を除いて他の部分をAVブロック内にフルにデータを書き込む形で記録を行えば、AVファイルのリアルタイム記録・再生を保証することが可能となる。というのはこれまで説明してきたように、1つAVブロックにフルに連続記録されたAVデータを読み出すと、ビットレート8Mbpsのデコードで1.5秒以上再生できるデータをトラックバッファに蓄積する事が出来る。この1.5秒は民生

のワーストケースのジャンプ時間であるから、データ読み出し中に如何なるジャンプが発生しても記録・再生を支障なく続ける事が可能となる。再生ではAVファイルの先頭の連続記録長がAVブロック長より短いと問題になりそうだが、トランクバッファにデータがフルになるまでデコードを開始しないとすれば、AVファイル再生の初期段階からジャンプを保証することが可能となり、再生が途切れるという問題は発生しない。このようにAVブロックとバッファリングモデルを用いることによりAVのリアルタイム記録・再生を保証しつつ、ディスクの使用効率を維持することができる。

【0104】なお、説明してきた様にバッファをフルにするためには、7.2MBの連続データは必要ない。

5.5MBのデータが連続的に書かれていればバッファをフルにすることは可能である。そのため、5.5MBのデータが連続的に書かれていればAVブロックに未使用領域を作るというAVデータの記録方法も考えられるが、ディスクにより多くのデータを記録するという立場から考えるとディスクの使用効率を下げるようなこの方法は利点がない記録方法となる。またAVファイルシステムが、AVブロックに連続的に5.5MBのデータを記録できたかどうかを管理せねばならず、AVファイルシステムの処理も複雑となってしまう。

【0105】(AVブロックとセクタビットマップの階層化) ここまでAVブロックのみに着目して説明してきたが、データ領域の割り当て状況を管理するという性質は、AVブロック管理テーブルも、セクタビットマップも共にもっており、データ領域を割り当てる場合は、AVブロック管理テーブルとセクタビットマップを同調して操作する必要がある。AVブロック管理テーブルはAVブロックが未使用(00)、AVファイル用(01)、非AVファイル用(10)のどの状態であるかを管理する。

【0106】またセクタビットマップはセクタの割り当て済み(1)、未割り当て(0)という状態を管理する。

【0107】ここでは、AVブロック管理テーブルとセクタビットマップの関係を説明する。ただし、実際の動作(READ、WRITEなど)に伴うAVブロック管理テーブルとセクタビットマップに対する処理の説明は、後でファイルシステムのコマンドを説明する際に行う。

【0108】AVブロックをAVファイル用に割り当てるとき、その中に含まれる全てのセクタはセクタビットマップ上に割り当て済みと登録される。図11においてAVブロックAV\_BLK#4がその例である。また図12(a)に、その場合のセクタビットマップの変化を示してある。実際にデータが書き込まれないセクタが有ったとしても、全て割り当て済みとなるのである。こうすることで、セクタビットマップのみをサポートし、AVブロック管理テーブルをサポートしないファイルシステムにこのディスクをアクセスさせても、DVD-RAMのファイルシ

ステムで書き込まれたAVファイルを保護することが可能なのである。

【0109】AVファイル用に割り当てられたAVブロックは、記録されている全てのAVデータが削除された場合、AVブロック管理表に未使用のAVブロックと登録される。具体的にはAVブロック管理テーブルの01の値が00に変更される。また同時に未使用に変更されたAVブロックに含まれるセクタも状態が変更され、セクタビットマップ上で未割り当てセクタと変更される。図12(b)にそのときの処理内容を示しておく。AVブロック内のAVデータが全て削除されたかどうかは、セクタビットマップから判断できないため、ファイルのExtent情報を元にAVファイルシステムが判断する。

【0110】AVブロックを非AVファイル用に割り当てるとき、AVブロック管理テーブルに10と登録される。図11のAV\_BLK#1はその例である。AVブロックを非AVファイル用に割り当てる場合、AVブロックに含まれるセクタは実際にデータが書き込まれるものに関してのみセクタビットマップを割り当て済みと変更される。

つまりAVファイルを記録するときのように、データを記録しないセクタまでセクタビットマップ上で割り当て済みにする事はないのである。こうすることで、非AVファイル用AVブロックでは複数の非AVファイルが存在することができ、ディスク全体の使用効率を改善することが可能となる。また、こうすることでセクタビットマップのみをサポートするファイルシステムでも、非AVファイル用AVブロックに非AVファイルの書き込みが可能となる。

【0111】更にセクタビットマップのみをサポートするファイルシステムで非AVファイルが書き込まれたディスクも、DVD-RAMのファイルシステムにかける際には、はじめにセクタビットマップをサーチして、AVブロック管理テーブルで、00となっているAVブロックに含まれるセクタであるのに、セクタビットマップ上で割り当て済みの状態をもつセクタが存在すれば、そのセクタを含むAVブロックを10としてAVブロック管理テーブルに登録することで、DVD-RAMのファイルシステム上でのディスクの正当性を保つ事が出来る。

【0112】AVブロックの記録方法に関して付け加えておくと、非AV属性のAVブロックに対しては、データの記録は代替セクタ方式で記録される。しかし、特に記録方式に対する制限はないので、別にECCブロックスキップ方式(アドレスエラーがあるECCブロックはスキップして次のECCブロックに記録を行う。特平8-258078参照)で、データを記録してもよい。ただし、同一のECCブロック中で先の2つの記録方式が混在することだけは避けなければならない。

【0113】以上がAVブロックの原理の説明である。(AVブロックを処理するAVファイルシステムAPI)図1に示してあるファイルシステムのコマンドのうち、

図24のDVDレコーダーの動作について説明する際に必要とされ、しかもAVブロック管理テーブルとセクタビットマップに関するコマンドをここで説明する。

【0114】「WRITE」は、ディスク上に非AVファイルを書き込む場合に実行される。「WRITE」はまずAVブロック管理テーブルを検索し、10の状態をもつAVブロックを探す。次に見つけたAVブロック内のセクタの状態をセクタビットマップより検索して、未割り当てのセクタがあればそこにデータを書き込み、データを書き込んだセクタをセクタビットマップ上で割り当て済みと状態を変更する。10の状態をもつAVブロック内に十分な空き領域がない場合は、もしくは10のAVブロックが存在しない場合は、00のAVブロックを新しく非AVファイル用のAVブロックとして割り当て、その中のセクタに非AVファイルを書き込む。この場合もセクタに対する処理は先と同様に行う。01の状態のAVブロックに対して非AVファイルを書き込む事は無い。先にも説明した様に、AVファイルシステムはファイルをExtentのリンクリストで管理するため、非AVファイルを書き込む際に記録するセクタが不連続になると、新しくExtentを生成することになる。

【0115】「AV-WRITE」は、ディスク上にAVファイルを書き込む場合に実行される。「AV-WRITE」は、AVブロック管理テーブルを先頭から00の状態のAVブロックを検索して、先にAVデータを記録していたAVブロックに連続した形でなるべく新しいAVブロックを確保しようとする。確保されたAVブロック内に存在する全セクタはセクタビットマップ上で割り当て済みと変更される。そして確保したAVブロックに対して先頭からAVファイルを書き込んでいく。「AV-WRITE」におけるAVブロック管理テーブルとセクタビットマップの変化の様子は図12に示して有る。「WRITE」とは逆に「AV-WRITE」は10の状態を持つAVブロックに関してAVファイルを記録することは無い。AVファイルシステムでは、ファイルはExtentのリンクリストとして管理される。AVファイルを書き込んでいる際に、AVブロックが不連続になった場合、新たなExtentが生成される。AVファイルをゾーンを跨いで記録する場合、ディスクのゾーンの境界にはバッファセクタが存在するため必ずゾーン境界でExtentが切れることになる。

【0116】「READ」は、ディスク上に記録されたデータを、指定されたサイズだけ読み出す場合に実行される。「READ」は一度に32KB程度のデータを読み出す。これはエラー訂正の最小単位ECCブロックが16セクターで構成され、それが32KBであることに依る。

「READ」が連続的にデータを読み出しているときに、Extentの境界に到達すると、次のデータが記録されているExtentまでジャンプを行い、そこからまたデータの読み出しを行う。このジャンプ時間はワーストケースで1.

5秒であるが、AVブロック内のセクターには連続的に

AVファイルが記録されていることから、AVブロック内の全セクターを連続的に読むことで、1.5秒の間AVデコーダーに提供するデータをトランクバッファに蓄積することが出来、AVファイル内でのシームレス再生を保証することができる。「READ」自体はAVブロック、セクタビットマップに対して処理を行うことは無い。

【0117】なお、「READ」は指定されたデータ内にAVブロックの境界がある場合に、カレントのヘッドの位置から近い方のAVブロックに記録されているデータを先に読み出すことで、データ読み出し時間の最適化を行うことも可能である。

【0118】「DELETE」はディスク上に存在するファイル全体を削除する場合に実行される。「DELETE」はファイルのExtentのリンクリストをたどりながら、全てのExtentを削除することでファイルの削除を実現する。「DELETE」をAVファイルに対して実行するのか、非AVファイルに対して実行するのかでAVブロック、セクタビットマップへの処理が変わってくるので、それぞれの場合について説明を行う。

【0119】まずAVファイルを削除する場合について図13を用いて説明する。図13はAVfile#2を削除する例である。AVfile#2はAV\_BLK#11とAV\_BLK#14に記録されており、AVブロック管理テーブル上でAV\_BLK#11, #14(斜線部)は01と登録されている。AVfile#2に対して「DELETE」コマンドが実行されるとAVブロック管理テーブルのAV\_BLK#11, #14の部分が01から00に変更され、未使用AVブロックとして管理される。併せて、AV\_BLK#11, #14に含まれる全てのセクタについてセクタビットマップを割り当て状態(各ビットが1)から未割り当て状態(各ビットが0)に変更する。

【0120】次に図14を用いて非AVファイルを削除する場合を説明する。削除する非AVファイルをfile#3とする。file#3は1つのExtentから構成されており、Extentの内容は図13に示すとおりである。今file#3を「DELETE」コマンドにより削除すると、file#3が記録されていたセクタsector#100からsector#110のセクタビットマップの内容(斜線部分)が割り当て状態から未割り当て状態に変更される。AV\_BLK#11のAVブロック管理テーブルの状態は、その時のセクタビットマップの状態により処理が変わってくる。file#3以外にもAV\_BLK#11内に非AVファイルが記録されている場合、AV\_BLK#11についてAVブロック管理テーブルを変更することはない。しかし、file#3を消すことでAV\_BLK#11内に存在するファイルが無くなる場合、AV\_BLK#11についてAVブロック管理テーブルを10から00に変更して、AV\_BLK#11を未使用状態にする。

【0121】「SHORTEN」は、AVファイルの端の要らない部分を削除する場合に実行される。「SHORTEN」はAVファイルのみに有効なコマンドで、非AVファイル

に対して実行するとエラーとなる。「SHORTEN」にはAVファイルの先頭部分を削除する場合と、AVファイルの末尾部分を削除する場合の2通りの使い方がある。

【0122】AVファイルの部分削除は、1つのAVブロック内で処理が完結出来るのであればExtent情報を変更する事で実現する。つまり実際にディスク上からデータを削除するわけではなく、Extentの情報を情報を変更することで、ファイルシステムから不要な部分のデータを見えないようにするのである。AVのAVブロックの場合、ファイルシステムはAVブロックの解放をExtent情報により判断して行う。なぜなら、AVブロックがAVファイル用に割り当てられると、セクタビットマップ上ではそのAVブロックに含まれる全セクタは割り当て済みとされてしまい、実際にデータが存在しないところで割り当て済みとなるので、セクタビットマップからAVファイルのデータが存在するところを判断することはできないのである。データを削除することで有効（ファイルシステムから認識される）データを全く持たないAVブロックが出来る場合には、そのAVブロックを未使用の状態に変更する処理を行う。この処理は「DELETE」の部分で説明したことと同様の事を行う。

【0123】図15を用いてAVファイルに「SHORTEN」を実行した場合の考え方を説明する。ただし、図15ではわかりやすいように、AVブロックからデータが実際に削除されたように書かれているが、先にも説明したように実際にはデータが残されていることに注意してほしい。図15(a)はAVfile#1の編集点より前の部分を削除する例である。「SHORTEN」を実行した事で、ファイルシステムからはAVfile#1の編集点より前のデータは見えなくなってしまう。図15(b)はAVファイルの末端の要らない部分を削除するために「SHORTEN」コマンドを実行する例である。図15(b)においてAVfile#1の編集点より前の部分を削除すると、AV\_BLK#kが有効なデータを持たないAVブロックとなる。この場合は先に説明した様に、AV\_BLK#kを未割り当て状態に変更する必要が出てくる。

【0124】「SPLIT」はAVファイルを2つに分割するために実行される。「SPLIT」はAVファイルにのみ有効なコマンドで、非AVファイルに実行するとエラーとなる。

【0125】図16を用いてAVfile#1を分割する例を説明する。ここでAVfile#1を分割した後でもAVfile#1という名前を使っているが、これは分割後の前のファイルが元ファイルの管理情報をそのまま受け継ぐためにそのようにした。「SPLIT」によってAVfile#1は（サイズ短くなった）AVfile#1とAVfile#3に分割される。この時編集点を含むAVブロックAV\_BLK#mには2つのファイルが存在することになる。AVブロックにAVデータを記録する場合、1つのAVブロックには1つAVファイルしか記録できないため、共存しているAVfile#3のデータは他

の未割り当て状態のAVブロックへ移動させなければならない。そのためAVブロック管理テーブルから00の状態のAV\_BLK#nを見つけ、状態を01に変更して、AVfile#3-1をAV\_BLK#mから移動する。AV\_BLK#nはAVファイル用のAVブロックとなるので、AV\_BLK#nに含まれる全てのセクタは割り当て済みとセクタビットマップを変更される。

【0126】「MERGE」はディスク上の2つのAVファイルと、メモリ上のデータを接続して1つのAVファイルを作る場合に実行される。「MERGE」におけるデータの移動の様子を図17を用いて説明する。図17はAVfile#1とAVfile#2とメモリ内のデータを、AVfile#1、メモリ内データ、AVfile#2という順番で接続する例である。この場合out点直前のAVfile#1の連続記録長がAVブロック長より長く、AV\_BLK#nのout点以降にメモリ内のデータサイズ以上の空き領域（ベクタビットマップには割り当て済みとされているが）があるため、そこにメモリ内のデータを書き込み、メモリ内のデータ最後からのジャンプ先をAV\_BLK#nのin点に設定すればmerge処理は終わる。

【0127】しかし、図17の(b)の場合、つまりメモリ内のデータサイズkがout点以降の空き領域のサイズiより大きい場合、空き領域にメモリ内のデータが書き込めないため、他の空き領域にAVfile#1のデータを移動させて、メモリ内のデータと合わせてデータ長がAVブロック長より長い連続記録データを作成する必要がある。これは、AVファイルの端部以外の連続記録データは必ずAVブロック長以上でなければならないと制約があるからである。AVブロックの長さを決定する考え方を示したところで説明したとおり、もしこの制約を満たさないとファイル内での連続再生を保証することができなくなってしまう。図17(a)はAVfile#1が既にこの条件を満たしていたので、データの移動を行わなかったのである。

【0128】AVファイルの連続記録データ長がAVブロック長以上でなければならないという制約は図18の様なAVfile#1-1のデータサイズlとメモリ内のデータサイズkの合計がAVブロック長以下の場合も適用されデータの移動が発生する。もしメモリ内のデータをAV\_BLK#nに書き込むとAV\_BLK#n内のデータ長がAVブロック長以下になってしまい、ファイル内での連続再生を保証できない。AVブロック長以上の連続データを作成するため、AVfile#1-1とAVfile#1-2を空きAVブロックが2つ連続する領域で2つにまとめ、その後ろにメモリデータを追加することによりAVブロック長以上の長さの連続記録データを作成する。これによりファイル内シームレス再生を保証することが可能になる。なお、AVfile#2のin点以降の連続データ長がAVブロック長未満の場合、これもファイル中の連続記録データ長がAVブロック長未満の場合に該当するので、データの移動を行い連続記

録データを再構成する必要がある。

【0129】なお、図17 (a)においてAVfile#2のin点直前の空き領域のサイズjがメモリ内のサイズkよりも大きい場合、図21に示す様にメモリ内のデータを、in点直前の空き領域にAVブロックの途中から書き込みという特別な記録の仕方を行い、ファイルをmergeする方法も考えら得る。これ以降にもAVfile#2の前にメモリ中のデータを書き込むという方法を幾つか説明するが、この場合メモリ中のデータをAVfile#2に連続する形で書き込む事に注目して貰いたい。空き領域の途中から書き込むことになるのも、このことが直接の原因である。というのはAVデータはなるべくデータを連続した形で記録し、余分なジャンプを同一AVブロック内では起こらない様にしなければならないのである。余分なジャンプがあるために場合によっては、記録されたAVデータの連続再生が出来ない場合が起きてくるのである。in点直前にメモリ内のデータを書き込めば、図17 (b)の時に余計なデータの移動は発生しない。もちろんメモリ内データとAVfile#2を結合して生成された連続記録データ長がAVブロック長以上であるという条件が必要である。

【0130】また、図17 (b)において空き領域iとjの合計がkより大きい場合、図22の (a) の様に、サイズiの空き領域にメモリ内のデータを書き込めるだけ書き、メモリ内の残りのデータをサイズjの空き領域内でAVfile#2につながる形で書き込むことが出来れば、余計なデータの移動を行う必要はない。但しこの場合も、生成される2つAVデータは、長さがAVブロック長以上の条件を満たす必要がある。また、メモリ内のデータの書き込む手順は先にin点の前に存在するサイズjの空き領域に書いてから、out点の後ろに存在する空き領域に残りのデータを書き込むという方法も考えられる。

【0131】なお、図22 (a) の方法を考えている場合であっても、iとjの合計がメモリ内のデータサイズより小さい場合は、図23に示す様にファイル1のデータの移動をしなくてはならなくなる。ただし、今回の場合は、図17 (b) の様にAVfile#1のデータをAVブロック分動かすことをせずに、メモリ中のデータをAVブロック長のデータにするために補う分のデータだけを移動させているのに注意して貰いたい。

【0132】また、図22 (b) の特殊なケースとして、図23に示すようにin点、out点に隣接する2つのデータが共にAVブロック長未満のサイズしかない場合がある。(a)の様に、in点の前、out点の後ろに空き領域があり、しかもメモリ中のデータをうまく分割すれば、両方のデータが共にAVブロック長以上のサイズを持てるのであれば、図23 (a) に示すようなメモリ中のデータの記入の仕方をすれば余分なデータの移動は起こらない。

【0133】また、in点、out点に隣接する空き領域がない (b) の様な場合であっても、in点とout点に隣接するデータとメモリ中のデータを足し合わせてAVブロック長以上のデータを構成できるのであれば、この3つのデータを、新たな空き領域を割り当て移動させれば、その他のディスク上のデータは移動させなくて済む。

【0134】色々な「MERGE」の方法を図で説明してきたが、いずれの場合も共通して言えることは最悪でも移動するデータサイズは2AVブロック長以下におさえることが出来ると言うことである。但し唯一の例外として、2つ連続した空きAVブロックが必要なのに、1つずつ孤立した空きAVブロックしか存在しない場合に、2つ連続した空きAVブロックを作るために、1AVブロックのデータを移動させることが必要となる。この場合にだけ余分な1AVブロックのデータの移動が発生し、全体として3AVブロックのデータの移動が起こることになる。

【0135】なお、「MERGE」は、1つのファイルとメモリ内のデータをmergeすることも可能である。図17においてAVfile#2を指定しなければ(NULLにすれば)、AVfile#1とメモリ内のデータのmergeを行うこと、つまりAVファイルの後ろにメモリ内のデータを追加することになる。また、AVfile#1を指定しなければ、AVfileの前にメモリ内のデータを追加する事になる。更にメモリのデータを指定しなければファイルのみのmergeも行うことが可能である。

【0136】これまでの話はファイル内のシームレス再生についてのみ考慮してきたが、ファイル間シームレス再生を考えると、これまで考えてきた「MERGE」の処理では不十分である。ファイル間シームレス再生を実現しようとする場合、AVファイルの記録で満足しなければならない条件が更に厳しくなる。ファイル内シームレス再生だけを保証するのであれば、一連のAVファイルを記録しているAVブロックの内、最初と最後のAVブロックを除く全てのAVブロックでデータの連続記録長がAVブロック長以上のサイズを持っていれば良かった。しかし、ファイル間シームレス再生を保証しようと、AVファイルを記録している全てAVブロックでAVデータの連続記録長でAVブロック長以上のサイズを持っている必要があり、図17 (b) や図18の様なデータの移動が、AVファイルの端部のにおいても必要となってくる。

【0137】「IN\_AV\_BLK\_BOUND」は、指定された2地点の間にAVブロックバウンダリがあるか判定する場合に実行する。もし指定された区間にAVブロックバウンダリが存在すればTRUEを返し、存在しなければFALSEを返す。図19においてa点とb点の場合は共にAV\_BLK#100の中にあるため、この2点を指定して「IN\_AV\_BLK\_BOUND」を実行するとTRUEが返される。a点とc点で定義される区間にはAV\_BLK#100とAV\_BLK#101の境界が存在す

るため、a点とc点を指定して「IN\_AV\_BLK\_BOUND」を実行するとFALSEを返す。このコマンドは特再の場合に意味を持つもので、詳しい説明は特再情報のところで行うためここでは省略する。

【0138】「SEARCH\_DISCON\_AV\_BLK」は指定された区間にAVブロック不連続境界があるか調べ、ある場合はTRUEを、無い場合はFALSEを返す。ここでAVブロック不連続境界の定義をしておく。AVブロック不連続境界とは、AVファイルの連続データを記録した2つのAVブロックの間に他のAVブロックを挟むか、もしくはゾーン境界を挟む状態を指す。

【0139】図20にAVブロック不連続境界の例を示す。(a)においてAVfile#1はAV\_BLK#mとAV\_BLK#nに連続するデータを記録している。AV\_BLK#mとAV\_BLK#nの間には複数のAVブロックが存在し、AVブロック不連続境界が発生している。Extentもここで分断される。

(b)の場合、AVfile#2はAV\_BLK#k、AV\_BLK#lに連続的に記録されている。しかし、この2つのAVブロックはそれぞれ異なるゾーン#i、#jに属しているため、このAVブロックの間にAVブロック不連続境界が発生してしまう。

【0140】「SEARCH\_DISCON\_AV\_BLK」は2地点を指定されて実行される。AVブロックが不連続である判定は、AVファイルのExtent情報から判断する。指定された2地点が同じExtentに含まれるのであれば、2地点間にAVブロック不連続境界は存在しない。指定された2地点が異なるExtentにそれぞれ含まれ、そのExtentの間にAVブロック長以上の隙間がある、もしくはExtentがそれぞれ違うゾーンに含まれる場合は、指定された2地点の間に必ずAVブロック不連続境界が存在することになる。

【0141】もう一つExtentが分割されるケースとしてECCブロックのスキップがある。AVファイルを記録する場合、アドレスエラーが起きたら、そのECCブロックをスキップして次のECCブロックにデータを記録するという記録方法(特平8-258078記載)を行った場合に、ECCブロックのスキップによりExtentの不連続が起きる。そのため、アドレスエラーによりECCブロックをスキップしてAVファイルを記録する場合も、Extentは分割されるが、この場合はAVブロック内でECCブロックの不連続境界であり、AVブロック不連続境界には該当しないとする。

【0142】ここまでに紹介したプリミティブなAVファイルシステムのAPIを用いて、AVデータを編集する場合を考えると、余分なデータの移動を発生することがある。

【0143】例えば、AVデータを分割するAPIに「SPLIT」というものが存在するが、AVデータの“AVデータは1つのAVブロックを占有する”という条件から、「SPLIT」で分割されることによって、編集点より後ろ

に生成されるファイルは、他の領域に移動されなければならない。しかし編集において「SPLIT」が単独で使われる事は少なく、その後に付随して行われる処理が存在する場合が多い。

【0144】例えばCMカットなどの不要部分の削除を行う場合、「SHORTEN」を実行するために不要部分の中でAVファイルを分割する必要がある。この場合にカットする領域内で「SPLIT」が実行され、新しく出来たAVファイルのデータは移動されてしまう。この後の流れの中で実行される「SHORTEN」によって削除される不要なデータであるにも関わらず移動してしまうと、余計なデータの移動が1回多く発生することになる。もし「SHORTEN」によって削除される不要な部分を取り除いた後に出来る2つのファイルが、それぞれ別々のAVブロックに存在していれば全くデータの移動は発生しないのである。この事を考えて、「SPLIT」から「SHORTEN」までの処理を1つのコマンドとして実装すれば、余分なデータの移動を減らすことが可能となる。

【0145】また、「SPLIT」の実行においては、処理されたデータがAVファイルの規定を満足しなくてもよいとすれば、余計なデータの移動は発生しない。これによりデータの移動量は削減出来るが、AVブロックの管理が多少複雑になる。AVファイルがAVブロックの一部にデータを持つ場合、それを削除したからといって、そのAVブロックをすぐに未使用に出来ない。なぜなら、AVブロックの残りの空き領域に、他のAVファイルのデータが存在するかもしれないからである。これは全てのファイルのExtentを調べれば分かるが、これを行うのは面倒である。これの解決策としては、AVブロック管理テーブルに記録されているAVファイルの開始アドレスとサイズを持たせるという方法が考えられる。また、AVファイルシステムしかこのディスクにアクセスしないとすれば、セクタビットマップをAV属性のAVブロックに関しても、データが記録されたセクタしか割り当て済みと登録しない様にすることで、AVブロック内のデータの存在を確かめることが可能となり、AVファイルは1つのAVブロックを占有するという制限を外すことが出来る。

【0146】4. プレゼンテーションデータ  
ここでは、MPEGストリーム(以下「プレゼンテーションデータ」とも呼ぶ)が収められているAVファイルについて説明する。

【0147】(ディレクトリ構成)図29はDVDレコーダにおいて必要なディレクトリとファイルの構成を示す。

【0148】DVDレコーダのファイルは全てROOTディレクトリの下のDVD\_RAM\_AVという専用のディレクトリに記録される。図中で拡張子が“.avi”であるAV\_File\_#1.avi,,,AV\_File\_#n.aviは実際のMPEGデータを記録したAVファイルを示す。これらのファイルは前述のAVブロックを用いて連続的に記録される。図中で拡張子が“.i

“.fo”は後述するAVファイル管理情報ファイル、“.cl p”は後述するAVクリップ管理情報ファイルのAV Clip partである。AVファイル、AVファイル管理情報ファイルおよびAVクリップ管理情報ファイル間では、拡張子を取り除いたボディー名で対応が取れているものである。また、ファイル名が固定である“Clip\_Sequence”は、後述するAVクリップ管理情報ファイルのAV Clip Sequenceである。

【0149】以降ここではAVファイルの論理フォーマットについて説明する。まず、最初にMPEGデータについて簡単に説明し、次に編集によってできるAVファイルのデータ構造および作成方法について説明する。

【0150】(ビデオデータ)まず、MPEGのビデオデータについて説明する。

【0151】MPEGのビデオ圧縮では、圧縮率を高めるため、フレーム内での空間周波数特性を利用した圧縮だけでなく、過去および未来からの時間相関特性を用いたデータ圧縮も行っている。図30を用いて説明すると、MPEGでは、過去および未来からの参照を行うBピクチャと、過去からの参照のみを行うPピクチャと、時間相関を用いないIピクチャの3種類のピクチャタイプが存在する。時間相関による圧縮を用いているPおよびBピクチャは参照先のピクチャの情報をもとにデコードされるので、これらのピクチャのデコードには参照先のピクチャがデコードされている必要がある。そのため、未来からの参照を行っているBピクチャは、参照先である未来のピクチャがデコードされた後でなければ、デコードができない。そこで、MPEGの圧縮データ(ストリーム)では、ピクチャの再生順(display order)と異なり、図30に示すようにBピクチャを参照先のピクチャの後に並べる(coding order)ことを行っている。また、この順番の入れ替えをリオーダと呼んでいる。

【0152】時間相関を用いたPおよびBピクチャは、参照先のピクチャを利用してデコードを行うので、単独でのデコードはできない。そのため、PおよびBピクチャのみを連続して使用すると、特殊再生などでストリーム途中からのデコードを行う場合に問題が生じるので、一般的には約0.5秒毎にIピクチャを入れることを行う。このIピクチャを先頭として、次のIピクチャ先頭までをGOP(Group of Pictures)と呼びMPEGでの一圧縮単位として扱っている。

【0153】また、MPEGでは、時間相関を用いた圧縮を行っている関係から、GOPではdisplay orderで一番最後のフレームがIまたはPピクチャで終わり、coding orderで一番最初のフレームがIでなければならない制限がある。

【0154】(多重化)次に、AVデータの多重化について説明する。

【0155】ビデオおよびオーディオデータは、約2KBの単位で分割され、2KB固定長のパックに収められる。

ビデオパックおよびオーディオパックは、デコーダバッファに入力する順に並べられ、パックヘッダにデコーダバッファへの入力時刻を示すタイムスタンプであるSCR(System Clock Reference)が付けられる。また、パック内のパケットヘッダ中には、ビデオおよびオーディオデータの再生時刻を示すPTS(Presentation Time Stamp)と、デコード時刻を示すDTS(Decoding Time Stamp)が付けられている。

【0156】このようにして多重化が行われるビデオ、オーディオデータでは、次のような特長がある。

【0157】MPEGビデオでは、高効率な圧縮を実現する為、画像の複雑さに応じた動的な符号量割り当てをフレーム単位でできるようにデコーダで入力バッファを持ち、このデコーダバッファへ予めデータを蓄えておくことで圧縮の難しい複雑な画像に対しての大量のデータ割り当てを可能としている。従ってデコーダバッファへ予めデータを蓄える為の時間(以下「VBV delay」と呼ぶ)だけビデオデータを早くデコーダバッファへ入力する必要がある。これに対してオーディオデータは各フレームとも固定サイズでエンコードされるため、ビデオデータのようにデコード時刻よりも特別に早くデータ入力をを行う必要はない。従ってビデオデータはオーディオデータよりも先行して多重化が行われる。

【0158】図31はビデオデータおよびオーディオデータの多重化の例である。図31は上から、(a)がビデオデータ、(b)がビデオバッファの状態、(c)がビデオデータとオーディオデータを多重化したAVデータ(夫々パック化され多重化されている)、(d)がオーディオデータを示している。横軸は各図に共通した時間軸を示していて、各図とも同一時間軸上に描かれている。また、ビデオバッファの状態において、縦軸はバッファ占有量(ビデオバッファのデータ蓄積量)を示し、図中の太線はバッファ占有量の時間的遷移を示している。また、太線の傾きはビデオのビットレートに相当し、一定のレートでデータがバッファに入力されていることを示している。また、一定間隔でバッファ占有量が削減されているのは、データがデコードされたことを示している。また、点線(斜め線)と時間軸の交点はビデオフレームのビデオバッファへのデータ転送開始時刻を示している。

【0159】以降、ビデオデータ中の複雑な画像Aを例に説明する。図31(b)で示すように画像Aは大量の符号量を必要とするため、画像Aのデコード時刻よりも図中のVBV delayだけ早い時間からビデオバッファへデータ転送を開始しなければならない。その結果、AVデータとしては網掛けされたビデオパックの位置(時刻)で多重化される。これに対して、オーディオデータの転送はデコード時刻より特別に早まらないで、図中の網掛けされたオーディオパックの位置(時刻)で多重化される。従って、同じ時刻に再生されるビデオデータとオ

一ディオデータでは、ビデオデータが先行している状態で多重化が行われる。尚、MPEGでは、バッファ内にデータを蓄積できる時間が限定されていて、全てのデータはバッファに入力されてから1秒以内にバッファからデコーダへ出力されなければならないように規定されている。そのため、ビデオデータとオーディオデータの多重化でのずれは最大で1秒（厳密に言えばビデオデータのリオーダの分だけ更にずれることがある）である。

【0160】尚、本例では、ビデオがオーディオに対して先行するましたが、理屈の上では、オーディオがビデオに対して先行することも可能である。ビデオデータに圧縮率の高い簡単な画像を用意し、オーディオデータを不必要に早く転送を行った場合に、このようなデータを意図的に作る事は可能である。しかしながらMPEGの制約により、先行できるのは最大でも1秒までである。

【0161】（編集後のAVファイル）次に、編集を行った結果、AVファイルがどのような構造に変化するのかについて説明する。

【0162】例として図26に示す3つのAVクリップを用いて説明する。まず編集を行う以前の段階では、AVファイル#j、#kとも図27に示す標準的なMPEGデータ形式で記録される。つまり、先頭のパックのSCRを0として時間的に連続するタイムスタンプが付与される。このようなMPEGデータをDVDではVOB（Video Object）と呼ぶ。VOBの正確な定義はISO/IEC13818-1で規定されるMPEGのプログラムストリームであって最後にprogram\_end\_codeが付かないものである。

【0163】3つのクリップを編集して一つのAVファイルを構成する場合、結果として3つのクリップに対応する3つのVOBから構成される一つのAVファイルが作成される。各VOBの境界ではオリジナルのVOBから切り出された位置に応じて先頭のタイムスタンプが異なるため、VOBの境界ではタイムスタンプは連続していない。

【0164】図32は、タイムスタンプが連続しない3つのVOBから構成されるAVファイルの例である。編集によって作られたVOB境界点ではタイムスタンプの不連続が生じる。MPEGで定義されているデコーダモデル（以下「STD」と呼ぶ）はタイムスタンプが連続するストリームに対しての動作を規定したモデルであるため、本例のようにタイムスタンプの不連続が生じると、VOB境界を途切れることなく連続再生（以下「シームレス再生」と呼ぶ）することができなくなる。

【0165】（E-STD）DVDでは、図33に示す拡張STDモデル（以下「E-STD」と呼ぶ）を定義し、タイムスタンプが連続しないVOB間のシームレス再生を実現する。E-STDは、MPEGで定義しているSTDに比べて、基準時間を刻むシステムタイムクロック（以下「STC」と呼ぶ）にオフセットを加える加算器とSTCの出力値と加算器の出力値の一方を選択できるスイッチが付加された構成となっている。この構成によって、VOB間で固有のオフセ

ット（「STC offset」と呼ぶ）を加えることで、図32に示すようにVOB間で擬似的にタイムスタンプを連続させることができとなる。STC offsetは次のようにして定義される。ストリーム中のタイムスタンプで表現したVOBのビデオ表示開始時刻をVOB\_V\_S\_PTと定義し、終了時刻をVOB\_V\_E\_PTと定義する。この時、STC offsetは次の式より求まる。

$$[0166] STC\ offset = \text{前部VOBのVOB\_V\_E\_PTM} - \text{後部VOBのVOB\_S\_PTM}$$

次に、E-STDの動作について簡単に説明する。E-STDの動作で重要なことは、図33中の各スイッチ（SW1、SW2、SW3およびSW4）の切り替えタイミングである。中でも、一番最初に切り替えを必要とするデマルチプレクサ用のスイッチSW1の切り替えタイミングは、他のスイッチに対しての切り替え予告となり得るので特に重要である。スイッチSW1の切り替えは、前部VOBの最後のパックの入力開始から後部VOBの最初のパックの入力開始までの間なので、前部VOBの最後のパックの入力開始時刻を示すSCR（LAST\_SCR）と、後部VOBの最初のパックの入力開始時刻を示すSCR（FIRST\_SCR）をSW1切り替えタイミングとしてE-STDに与えれば良い。これによって通常再生時のスイッチ切り替えが可能になる。

【0167】また、図33中のADPI（Audio Decoder Pause Information）は、オーディオデコーダを一時的に停止させるための情報であるが、詳しくは、後述するオーディオギャップで説明する。

【0168】尚、E-STDの基本動作については、特許「国際公開番号WO97/13364」で詳しく述べているので、本例でのこれ以上の説明は省略する。

【0169】（オーディオギャップ）VOB間でのシームレス再生を実現するために幾つかの条件がある。一つ目はオーディオギャップである。DVDで扱うAVデータは、ビデオの場合はNTSC、PALであり、オーディオの場合はDolby AC3、MPEG、LPCMである。これらのAVデータは全てフレームという概念を持っていて、夫々異なるフレーム周期で構成されている。具体的には、NTSCの1フレームは約33 msec（正確には1/29.97 sec）、PALの1フレームは40 msecであり、Dolby AC3の1フレームは32 msec、MPEGの1フレームは24 msec、LPCMの1フレームは約1.67 msec（正確には1/600 sec）である。このため、どのビデオ、オーディオの組み合わせにおいてもフレーム周期が異なるため、AVファイルからVOBを切り出して、異なる2つのVOBを繋ぎ編集を行う場合、ビデオを基準にVOBを繋ぐとオーディオデータに再生のギャップまたはオーバーラップが生じ、オーディオを基準にVOBを繋ぐとビデオデータに再生のギャップまたはオーバーラップが生じてしまう。DVDでは、ビデオを基準としてオーディオのギャップが生じるものとする。

【0170】図34は、本例のオーディオギャップを示した図である。MPEGデータにはデコード時刻や再生時刻

を示すタイムスタンプが付加されているが全てのフレームに対して付けられているわけではない。例えば、ビデオデータの場合、1GOPに一回、即ち約0.5秒に一回つけられることがしばしばある。そこでデコーダでは、AVデータが一定のフレームレートであることを利用して、一定周期で連続的にAVデータのデコードを行っている。このようなデコーダにオーディオギャップを有するAVデータを入力する場合、デコーダに対してシステム制御部(図24の録画・編集・再生制御部)からオーディオポーズを指示してオーディオデコーダを一時的に停止する必要がある。これが図33中のADPI(Audio Decoder PauseInformation)である。

【0171】このようにして、オーディオギャップを処理することが可能となるが、汎用的なマイコンとソフトウェアから構成されるシステム制御部からオーディオデコードの停止を制御する都合上、オーディオギャップが短期間に連続しては、システム制御部の処理が間に合わない問題が生じる。そこで、オーディオギャップが一定の間隔以上ひらくように制限を設ける必要がある。

【0172】本例では、以下の制限を設け上記の問題を回避することとする。オーディオギャップは編集により生じるものであるから、オーディオギャップは編集の単位、即ちVOBに対して最大1つとする。次に、オーディオギャップの時間長に制限を設け、1オーディオフレーム未満とする。次に、オーディオギャップのにおける位置(時刻)に制限を設ける。オーディオギャップは編集により生じるものであるから、編集の境界面であるVOBのビデオ表示開始時刻を基準として、前後1オーディオフレーム未満の間にオーディオギャップの開始時刻を置くように制限する。次に、VOBの時間長を1.5秒以上に制限する。

【0173】以上の制限を設けることで、シームレス再生時におけるオーディオギャップの発生間隔は最小でも、"1.5秒-1オーディオフレーム再生時間×2"に収めることが可能である。具体的に数値を当てはめてみると、オーディオをDolby AC3とすれば1オーディオフレーム再生時間は32msecであるから、オーディオギャップの発生間隔は最小でも1436msecになる。

【0174】(バッファ制御)二つ目はVOB間でのバッファ制御の連続性である。

【0175】前述した多重化の説明でも述べたが、MPEGは高効率な圧縮を実現するためにデコーダに入力バッファを設けて、各フレームに必要な符号量を動的に割り当てられるようにしている。この反面、デコーダバッファがアンダーフローまたはオーバーフローを起こすと正常な再生が行えず、再生途中で途切れるなどの障害が生じてしまう。そこでMPEGのエンコーダでは、デコーダバッファがアンダーフローまたはオーバーフローを起こさないようにエンコーダ内部に仮想バッファを設けてデコーダバッファのシミュレーションを行っている。

【0176】このような一連の流れの中でエンコードされたMPEGデータはデコーダバッファのオーバーフローまたはアンダーフローが起きないことが保証されるが、別々にエンコードされたVOB間でシームレス再生を行う場合、VOB間でデコーダバッファがアンダーフローまたはオーバーフローしないことを保証できない。そのため、VOB間でシームレス再生を行うには、VOB間でのバッファ制御の連続性を保証する必要がある。そこで、編集時にVOB間でのバッファ制御が連続するようにVOB境界近傍を局所的に再エンコードを行う。

【0177】(再エンコード方法)次に、VOB境界近傍での局所的な再エンコードについて説明する。

【0178】まず、VOB境界となるAVデータを読み出す。読み出し方法、後述する編集の動作で説明する。読み出したAVデータを、ビデオ、オーディオ夫々に分離し、夫々に処理を施す。

【0179】まず、ビデオのGOP再構成を行う。ビデオの再構成について図35を用いて説明する。

【0180】ビデオデータはGOPを一塊にした単位でエンコードが行われるため、編集によってGOPが崩されると、全てのフレームのデコードを正常に行うことができなくなってしまう。そこで、図35のようにGOP途中のフレームで編集が要求された場合は、GOPを新たに作り直す必要がある。先にも述べたが、MPEGには、時間相関を用いないIピクチャと、過去からの時間相関を用いたPピクチャと、過去および未来からの時間相関を用いたBピクチャの3種類のピクチャタイプがあり、MPEGのBピクチャと、IおよびPピクチャでは、フレームの表示順(display order)と、フレームの圧縮データ順(coding order)とが入れ替わっている。また、GOPでは、フレームの表示順(display order)で、一番最後のフレームがIまたはPピクチャで終わり、フレームの圧縮データ順(coding order)で、一番最初のフレームがIでなければならないため、GOPを再構成する場合、以下のルールに従った処理が必要となる。

【0181】一つ目は、前部のVOBの最後のGOPにおいて、フレームの表示順(display order)で、最後のピクチャがBピクチャの場合は、このピクチャをPピクチャ(またはIピクチャ)にピクチャタイプを変更する(ルール1)。

【0182】二つ目は、後部のVOBの最初のGOPにおいて、フレームの圧縮データ順(coding order)で、最初のピクチャがPピクチャの場合、Iピクチャに変更する(ルール2)。

【0183】三つ目は、後部のVOBの最初のGOPにおいて、フレームの表示順(display order)で、最初のIピクチャ(ルールに2に従い変更されたIピクチャも含む)より先に表示されるBピクチャでは、過去からの時間相関を打ち切るようにピクチャ内の圧縮方法を変更する(ルール3)。

【0184】次に、ビデオバッファ制御を行う。ビデオバッファ制御について図36を用いて説明する。

【0185】バッファ制御の説明でも述べたが、編集を行う2つのVOB間でビデオバッファ制御が連続しなければならない。そこで、2つのVOB間でのバッファ制御が連続できるようにVOB境界近傍での再エンコードを行う。図36は、再エンコード時のバッファ制御を示した図である。まず、VOB境界でのバッファ占有量Bv1を決める。Bv1の決めかたに限定はないが、本例では、後部VOBでの再エンコード前のバッファ占有量をそのまま用いる。前部VOBでは、前部VOB最後(VOB境界)のバッファ占有量がBv1となるように、各ピクチャでの符号量割り当てを行い直し、再エンコード区間の各ピクチャの再エンコードを行う。後部VOBでは、VOB先頭でのバッファ占有量をBv1として、再エンコード範囲後の最初のピクチャでのバッファ占有量Bv2(再エンコード前と同一値)に繋がるように、再エンコード区間でのバッファ制御を行う。

【0186】以上の処理によって2つのVOB間でのバッファ制御は連続させることが可能となる。また、ビデオの再エンコードを行う範囲は、少なくともピクチャタイプが変更されるフレームは必要であるが、この限りではなく、複数VOBUに跨って再エンコードを行っても構わない。しかしながら、再エンコードの範囲が長くなるだけ編集時の再エンコード時間が長くなるデメリットが生じる。

【0187】次に、オーディオギャップの作成を行う。オーディオギャップの作成方法について図36を用いて説明する。

【0188】オーディオギャップの説明でも述べたが、ビデオとオーディオのフレームレートの違いによって、編集時にVOB境界でオーディオの不連続点(オーディオギャップ)が発生する。また、オーディオギャップの開始時刻は、VOBのビデオ表示開始時刻VOB\_V\_S\_PTの前後1オーディオフレーム未満になければならない。そこで、次のようにしてオーディオギャップを決定する。図36において、時刻t2は後部VOBのVOB\_V\_S\_PTであり、この時刻t2を規準として、前部VOBでは、時刻t2を含むオーディオフレームまでを編集後のデータとして使用し(時刻t2がオーディオフレーム境界に一致する場合は、時刻t2で再生終了となるフレームまでを使用する)、後部VOBでは、前部VOB最後のオーディオフレームの再生終了時刻t3直後(時刻t2を含む)に再生開始する最初のオーディオフレームからを編集後のデータとして使用する。

【0189】また、前部VOB最後のオーディオフレームの再生終了時刻t3から、後部VOB最初のオーディオフレームの再生開始時刻t4までの時間がオーディオギャップ長A\_GAP\_LENになり、後部VOB最初のオーディオフレームの再生開始時刻t4からオーディオギャップ長A\_GAP\_LENだけ前の時刻がオーディオギャップ開始時刻A\_STP\_PT

になる。

【0190】次に、ビデオ、オーディオの多重化を行う。ビデオ、オーディオの多重化について図36を用いて説明をする。

【0191】前述した、ビデオのGOP構造の再構成、バッファ制御とビデオ再エンコード、オーディオギャップの作成を行った後、ビデオ、オーディオの多重化を行う。多重化の説明でも行ったがMPEGの性質上、ビデオデータはオーディオデータより先行して多重化が行われる。これは、VOB境界においても同様であり、後部VOBの先頭では、ビデオデータが先行して多重化される分だけ前部VOBにあったオーディオデータが後部VOBに回り込み多重化される。逆に、前部VOBでは、後部VOBに回り込んだオーディオデータを除いて多重化される。具体的には、図36の時刻t1(後部VOBでのビデオデータ入力開始時刻)を規準として、時刻t1直後(時刻t1は含まない)に再生が開始されるオーディオフレームからは、後部VOBとして多重化を行う。

【0192】尚、VOB間でのバッファ制御の連続および多重化方法については、特許「国際公開番号WO97/13367」および「国際公開番号WO97/13363」で詳しく述べているので、本例でのこれ以上の説明は省略する。

【0193】このようにシームレス接続・再生の考え方を用いると本来は全てのタイムスタンプを書き換えなければ編集できないMPEGデータであっても、大量のデータの移動や書き換えを伴うことなく簡易に編集が可能となる。

【0194】(非シームレス接続)次に、非シームレス接続について説明する。

【0195】編集によってAVファイル内に複数のVOBができるることは先に説明した。ここでは、AVファイル内のVOB間での非シームレス接続について説明する。

【0196】VOB間でのシームレス再生方法と、シームレスストリーム作成方法について前述したが、編集を行う全てのMPEGストリーム間をシームレスストリーム化することはできない。

【0197】例えば、編集により接続する2つのオーディオストリームが、一方がAC3のストリームであって、他方がMPEGのストリームである場合に、その接続境界に對してシームレスストリーム処理を行ったとしても、一般的にデコーダでシームレス再生はできない。ストリームがAC3からMPEGへ変化する場合、デコーダ内部でストリーム属性の切り替えを行うため、この間デコードが停止してしまうからである。オーディオストリームの属性が変わる場合も同様である。

【0198】以下の条件を少なくとも1つ満たす場合、シームレス再生はできなくなる。

1) ビデオのフレームレート(NTSC、PAL、...)が異なる。

2) オーディオエンコード方式 (AC-3、MPEG、LPCM) が異なる。

3) オーディオストリームのビットレートが異なる。  
【0199】また、意図的に非シームレス接続のVOBとすることも可能である。上記の場合、VOB間が非シームレスストリームとなるように編集を行う。具体的には、次のようにして再エンコードを行う。最初のVOB境界の読み出し、AVストリームの分離、ビデオのGOP再構成までは同一である。VOB間でのビデオバッファ制御は必要なく、各VOB単体でのビデオバッファ制御さえ行えば良く、オーディオデータの切り出しにおいても、オーディオギャップの作成のように厳密な計算は要求されず、ビデオにほぼ一致するオーディオデータを切り出せば良い。また、ビデオ、オーディオの多重化も各VOB内でのビデオ、オーディオの多重化さえ行えば良い。

【0200】5. ナビゲーションデータ

#### 5. 1. AVファイル管理情報

AVファイル管理情報の論理フォーマットについて図5 5と図5 7を参照しながら説明する。

【0201】図5 5はAVファイル管理情報の全体を示し、図5 7はAVファイル管理情報の一部であるStream Informationを示す。

【0202】(AV File Management Information)AV File Management Informationには、対応するAVファイルの識別名を示すAV File ID、そのAVファイルのセクタ単位の大きさを示すSector Size、そのAV Fileの再生時間を示すPlayback Timeと、VOB Tableが記録される。

【0203】VOB Tableには、AVファイルに含まれるVOBの数と、VOB Informationの配列へのポインタが記録される。VOB Informationには、AVファイル先頭からの相対セクタアドレスでVOB先頭のアドレスを示すStart Sector、AVファイルの先頭をタイムコード00:00:00:00としてVOBの先頭の再生経過時間を示すStart Time、Time Mapに記録されたStart Sectorの値を補正するためのSector Offset、Time Mapに記録されたStart Timeの値を補正するためのTime Offset、VOBのデータ属性を示すStream Informationと、次に説明するTime Map TableとVOBU Map Tableが記録される。

【0204】Time Map Tableには、Time Mapの数、Time Map間の時間間隔を秒で示すTime Unit、各Time Mapが指す時間を補正するためのTime Base、Time MapのVOBU Map Indexを補正するためのIndex Offsetと、Time Mapの配列へのポインタが記録される。Time Mapには、Time Mapが示す時刻に対応するVOBUのVOBU Map配列内でのインデックスを示すVOBU Map Index、VOBの先頭からのそのVOBUの先頭の相対セクタアドレスを示すStart Sector、VOBの先頭からのそのVOBU先頭のビデオフレームの相対タイムコードを示すStart Timeが記録される。

【0205】VOBU Map Tableには、VOBに含まれるVOBUの数とVOBU Mapの配列へのポインタが記録される。VOBU

10

20

30

40

Mapには、VOBUの先頭の相対セクタアドレスを示すStart Sector Offset、VOBU先頭のビデオフレームの相対タイムコードを示すStart Time Offset、早送りや巻き戻しなど特殊再生時に用いるリファレンスピクチャのVOBU先頭からの相対エンドアドレスを示すEnd Sector of the Reference Pictureが記録される。

【0206】Time Map Tableは、VOB先頭から一定間隔ごとの時刻に存在するVOBUの情報を記録する。VOBU Map Tableは、VOB中に存在する全VOBUの情報を、VOB中に存在する順番に従って、VOBU Map#1から順に記録する。

【0207】Time Map#iは、VOB先頭からの再生経過時間が((Time Unit) \* i + (Time Base))の時点に存在するVOBUを指す。Time Unitが1、Time Baseが0の場合には、Time Map#1、Time Map#2、Time Map#3は、それぞれ、VOBの先頭から1秒後、2秒後、3秒後に存在するVOBUを指す。Time MapのVOBU Map IndexとTime Map TableのIndex Offsetを足した値がVOBU Mapのインデックスを示す。Time MapのStart SectorとTime Map TableのSector Offsetを足した値が、Time Mapが指すVOBUの先頭のアドレスをVOB先頭からのセクタ単位のオフセットで表す。Time MapのStart TimeとTime Map TableのTime Offsetを足した値が、Time Mapが指すVOBUの先頭の再生経過時間を、VOBの先頭のタイムコードを00:00:00:00としたタイムコードで表す。

【0208】VOBU MapのStart SectorとStart Timeは、Time Mapで指されたVOBUまたはVOBの先頭を基準とした相対値で表す。Time Map#1が指すVOBUより前にあるVOBUに関しては、VOBの先頭を基準とする。その他のVOBUに関しては、Time Mapで指されたVOBUのうちで最も手前にあるものを基準とする。Time Unitを数秒に設定した場合は、Start Timeは2バイト、Start Timeは1バイトで表すことができる。

【0209】図5 6に、Time MapのVOBU Map IndexとVOBU Mapの関係を示している。

(Stream Information)Stream Informationには、ビデオストリームの種々の属性を示すVideo Attribute、オーディオストリームのマッピングを示すAudio Map Table、オーディオストリームの属性を示すAudio Attribute Tableを記録する。

【0210】Audio Map Tableは、8ストリーム分のAudio Mapを記録する。Audio Mapは、ストリームが有効か否かを示すValidity FlagとAudio Attribute TableのAudioAttributeを指すインデックスを記録する。

【0211】Audio Attribute Tableは、8ストリーム分のオーディオストリームの種々の属性を記録する。Video FormatはMPEG2の他にMPEG1も設定可能である。Video SystemはNTSCまたはPAL/SECAMが設定可能であり、NTSCではVideo Resolutionは他に352 x 480, 352 x 240が設定可能である。Video Aspectは4:3または16:9のワイドが設定可能であり、ワイドの場合における表示モード

として標準以外にLetter BoxまたはPan Scanが設定できる。Video APSはアナログビデオ信号用のコピー防止制御の方法を示し、AGCはビデオ信号のブランク区間の信号振幅を変化させVTRのAGC回路に障害を与えることによりVTRへのコピーを防止するものである。他にColor Stripeや両方の組み合わせも設定可能である。Audio Formatはオーディオのエンコード形式を示し、MPEG2やDolby Digital、リニアPCMなどを設定可能である。Audio Samplingはサンプリング周波数を示し、他に44.1KHzなどが設定可能である。Audio Bitrateは固定ビットレートの場合に値を示し、可変ビットレートの場合は“VBR”と記述される。これらのAttributesは図24中のAVデコーダ部131の初期設定に用いられる。

【0212】(AVファイル管理情報の生成)録画時のAVファイル管理情報の生成について説明する。

【0213】図24のAVデータ入力部111は、録画中にリアルタイムにGOPごとのセクタ数とリファレンスピクチャのセクタ数に関する情報をAVファイル管理情報生成部112に渡す。

【0214】AVファイル管理情報生成部112は、今までに生成したAVファイル管理情報と渡されたGOPの情報から、リアルタイムに現時点までのAVファイル管理情報を生成し、主記憶上に記憶する。そして、録画の終了時に、この主記憶上に記憶しているAVファイル管理情報を、非AVファイルとして、AVファイル管理情報ファイルに記録する。

【0215】なお、AVファイル管理情報生成部は、リアルタイムにGOPの情報をのみを記憶し、AVファイル管理情報をディスクに記録する前に、AVファイル管理情報を生成するという動作でも良い。

【0216】また、AVファイル管理情報生成部は、録画の終了前でも、AVデータのリアルタイムの記録を妨げないならば、AVファイル管理情報を、AVファイル管理情報ファイルに記録しても良い。

【0217】図58は、録画の開始から終了までの間に、AVファイル管理情報生成部が渡されるGOPの情報の例を示す。図59は、図58のGOPの情報から生成されるAVファイル管理情報を示す。また、図66は、3本のオーディオストリームが存在する場合のAVファイル管理情報の中のStream Informationの例を示す。

【0218】まず、録画後に生成されるAVファイル管理情報について説明する。図59は、図58のGOPの情報の例から生成されるAVファイル管理情報の例を示す。図59で示されたAVファイル管理情報の内容について説明する。VOBの数は1である。VOB InformationのSector OffsetとTime Offset、Time Map TableのTime BaseとIndex Offsetは、いづれも0であり、Time Mapに記録された値は、補正することなく、VOB先頭を基準にした値を示している。

【0219】次に、図66に示したStream Information

の内容について説明する。Audio Map及びAudio Attributeの最初の3つがValidとなり、その他はInvalidとなる。Audio MapのAudio Attribute Indexには、Audio Mapのインデックスと同じ値が記録される。

【0220】(AVファイル管理情報のSplit)編集時のSplitに伴うAVファイル管理情報の更新について説明する。図24の特再情報編集部が、Split要求を受けたときの動作について例を用いて説明する。図60は、Split前のAVファイル管理情報を示す。図61は、このAVファイル管理情報をVOB#3とVOB#4の間でSplitした後の2つのAVファイル管理情報を示す。

【0221】まず、前半部のAVファイル管理情報の内容について説明する。前半部のAVファイルが、元のファイルを引き継ぐため、AV File IDはSplit前と同じである。ファイルの大きさ、再生時間、VOBU数が、小さくなることに伴って、AV File Management InformationのSector SizeとPlayback Time、VOBU Map TableのVOBU数、Time Map TableのTime Map数が、小さくなる。

【0222】次に、後半部のAV管理情報ファイルの内容について説明する。AV File IDには新規の名前が記録される。ファイルの大きさ、再生時間、VOBU数が、小さくなることに伴って、AV File Management InformationのSector SizeとPlayback Time、VOBU Map TableのVOBU数、Time Map TableのTime Map数が、小さくなるのは、前半部と同様である。ファイルの前側がなくなるため、各VOBU Mapや各Time Mapに記録された値を補正しなければならない。そのために、各VOBU Mapや各Time Mapに記録された値を書き換えるのではなく、VOB InformationのSector OffsetとTime Offset、および、VOBU Time TableのTime BaseとIndex Offsetに、補正をするための値を設定する。

【0223】Stream Informationに関しては、Split前の値が、そのまま、Split後の両方のStream Informationに引き継がれる。

【0224】(AVファイル管理情報のShorten)なお、編集時のShortenの場合は、Splitによってできる2つのAV管理情報の一方のみが生成される動作と同じであるため、説明を省略する。

【0225】(AVファイル管理情報のMerge)編集時のMergeに伴うAVファイル管理情報の更新について説明する。図24の特再情報編集部が、Merge要求を受けたときの動作について例を用いて説明する。図62は、Merge前の2つのAVファイル管理情報を示す。図63は、この2つのAVファイル管理情報を間に再エンコードした2つのVOBUを追加してMergeした後のAVファイル管理情報を示す。ただし、前方のAVファイルの後には12フレーム40セクタからなるVOBUを、後方のAVファイルの前には18フレーム56セクタからなるVOBUを追加したものとする。また、図67はMerge前の2つのストリーム情報の例を示し、図68はMerge後の2つのス

トリー情報の例を示す。

【0226】MergeされたAVファイル管理情報の内容について説明する。前方のAVファイルにMerge後のAVファイルが引き継がれるため、AV File IDには前方のAVファイルの名前が記録される。AV File Management InformationのSector SizeとPlayback Timeは、2つのAVファイルと追加される2つのVOBUの分を合計した値となる。2つのAVファイルが、それぞれVOBとなるため、VOBの数が2となる。VOB#1に関しては、VOBU#5が追加されてVOBが大きくなることに伴って、VOBUMap TableのVOBU数とTime Map TableのTime Map数が大きくなる。VOB#2に関しては、VOBUが追加されてVOBが大きくなることに伴って、VOBU Map TableのVOBU数が大きくなる。前方にVOBとVOBが追加されたことにより、各VOBU Mapや各Time Mapに記録された値を補正しなければならない。そのために、各VOBU Mapや各Time Mapに記録された値を書き換えるのではなく、VOB InformationのSector OffsetとTime Offset、および、VOBU Time TableのTime BaseとIndex Offsetに、補正をするための値を設定する。

【0227】ストリーム情報に関しては、Audio MapにおけるValidなもののインデックスが同じであり、かつ、同じインデックスのAudio Mapに対応するAudio Attributeの属性が同じになるように、後半のAVファイルのStream Informationを変更する。

【0228】(AVファイル管理情報の大きさ)図5-9の例では、VOBU MapのStart Sectorが2バイト、Start Timeが1バイト、End Sector of the Reference Pictureが1バイトであり、Time MapのStart Sectorが4バイト、Start Timeが4バイト、VOBU Map Indexが2バイトである。例えば、再生時間が7時間で各VOBUが0.5秒のAVデータを記録するならば、VOBU Mapの大きさは約200キロバイトであり、また、Time Map TableのTime Unitを5秒とするならば、Time Map Tableの大きさは、約50キロバイトであるので、AVファイル管理情報の大きさは、約250キロバイトである。32キロバイトずつディスクにアクセスするならば、約8回のアクセスでAVファイル管理情報をディスクに書き出すことができる。

【0229】特殊再生を行う場合に、VOBU Map Tableを参照しないで、Time Map Tableのみを参照しても、特殊再生をすることが可能である。ただし、特殊再生の速度の種類が少なくなる。

【0230】このような特殊再生の方法により、主記憶の容量が少ない再生装置でもVOBU Map Tableを主記憶に記憶しないで、特殊再生を実行することができる。

【0231】5. 2. AVクリップ管理情報

(AVクリップ管理情報の論理フォーマット)次に、AVクリップ管理情報の論理フォーマットについて図3-7を参照しながら説明する。

【0232】まず、AVファイル毎に作られるAVクリ

ップ管理情報の内のAV Clip partについて説明する。

【0233】AV Clip partには、対応するAVファイルの識別名を示すAV File IDと、VOB毎に定義されるClip Informationの数と、Clip Informationへのポインタが記録される。Clip InformationはVOBを識別するVOB IDと、AVファイル先頭からの相対値をタイムコードで示したVOBのビデオ表示開始フレームStart Timeと、ストリーム中のタイムスタンプで示したVOBのビデオ表示開始時刻VOB\_V\_S\_PTと、ストリーム中のタイムスタンプで示したVOBのビデオ表示終了時刻VOB\_V\_E\_PTと、VOB先頭パックのバッファへの入力時刻であるFIRST\_SCRと、VOB最後のパックのバッファへの入力時刻であるLAST\_SCRと、直前のVOBとのシームレス接続情報を示すSeamless Informationと、Mark Tableと、Clip Tableが記録される。Seamless Informationには、シームレス接続または非シームレス接続するかを判別するSeamless Flagと、オーディオギャップの開始時刻をストリーム中のタイムスタンプで示したA\_STP\_PTと、オーディオギャップの時間長を示したA\_GAP\_LENが記録される。Mark TableにはMarkの数と、マークへのポインタが記録され、各マークにはVOB先頭からの相対値で示したタイムコード(Time\_Code)と、マークの指示するフレームが属するVOBUのVOB内の相対タイムコード(VOBU\_Time\_Code)と、このVOBUのVOB内の相対アドレス(VOBU\_Address)と、このVOBU先頭からこのフレームの最初のデータまでの相対アドレス(Frame\_Address)と、マークの削除の可否を示すDelete\_Permission(デフォルトは削除可)が記録される。Clip TableにはClipの数と、クリップへのポインタが記録され、各クリップにはStart Markと、End Markと、このクリップを一回以上再生したかを示すPlayedフラグ(録画時は未再生)と、クリップの削除の可否を示すDelete\_Permission(デフォルトは削除可)と、DVD-RAM上の空き領域が少なくなった場合に、自動的に削除しても良いことを示すAuto\_Deleteフラグ(デフォルトは自動削除不可)が記録される。

【0234】次に、AVクリップ管理情報の内のAVクリップシーケンス情報である、AV Clip Sequenceについて説明する。

【0235】AV Clip Sequenceには、Clip Sequenceの数と、Clip Sequenceへのポインタが記録される。Clip Sequenceには、クリップシーケンスに設定されたクリップを示すEntry Clipの数と、Entry Clipへのポインタとクリップシーケンスの削除の可否を示すDelete\_Permission(デフォルトは削除可)が記録される。Entry Clipには、クリップの実AVデータのあるAVファイルを識別するAV File IDと、VOBを識別するVOB IDと、Clipを識別するClip IDと、前のEntry Clipとの接続がシームレス再生されるかを判別するSeamless Flagが記録される。

【0236】(AVクリップ管理情報の作成)録画時に作

成されるAVクリップ管理情報について説明する。

【0237】図24のAVクリップ管理情報生成部は、録画終了時に図38に示すAVクリップ管理情報を生成し、ファイルに書き込む。次に、録画時に生成されるAVクリップ管理情報の内容について説明する。基本的に一回の録画では、1つのVOBと、2つのマークと、1つのクリップと、1つのクリップシーケンスが生成される。2つのマークは、VOBのビデオ表示開始および終了フレームであり、VOB先頭からの相対値である。1つのクリップは、この2つのマークをStart Mark、End Markとした1つのClipであり、1つのクリップシーケンスは、この1つのクリップから構成されたClip Sequenceである。各フィールドはAVクリップ管理情報のフォーマットで説明した通りに値が入れられる。

【0238】なお、AVデータ録画部の説明で行ったように、一時停止が行われた場合には、一時停止をした箇所にマークが置かれる。また、一時停止等によってVOBが分けられたとした場合、上記の通りVOB単位でマーク、クリップおよびクリップを作成するが、クリップシーケンスに限り、一連の録画単位となる1つのAVファイルで1つを作成する。

【0239】次に、マーク、クリップ、クリップシーケンスの追加方法について説明する。図24の録画・編集・再生制御部がマーク追加要求を受けた場合の処理について例を用いて説明する。図38は録画直後でAVクリップ管理情報を追加する前の図であり、図39はAVクリップ管理情報を追加した後の図である。

【0240】まず、マークの追加方法について説明する。図39は新たなマーク00:00:01:00を追加した場合のAVクリップ管理情報を示す。追加するマークは、Mark Tableで使用されていないID Mark#3が与えられ、Mark Tableに追加される。

【0241】次に、クリップの追加方法について説明する。図39は先に追加したマーク(図39のMark#3)を開始マークとし、AVファイル末尾のマーク(図39のMark#2)を終了マークとする新たなクリップを追加した場合のAVクリップ管理情報を示す。追加するクリップは、Clip Tableで使用されていないID Clip#2が与えられ、Clip Tableに追加される。

【0242】次に、クリップシーケンスの追加方法について説明する。図39は先に追加したクリップ(図39のClip#2)を一つの再生シーケンスとする新たなクリップシーケンスを追加した場合のAVクリップ管理情報を示す。追加するクリップシーケンスは、AV Clip Sequenceで使用されていないID Clip\_Sequence#2が与えられ、AV Clip Sequenceに追加される。追加されたクリップシーケンスは、先に追加したクリップ(図39のClip#2)から構成されるので、このクリップが属するAV File ID、VOB ID、Clip IDと、Seamless FlagがEntry Clipとして管理記録される。

【0243】次に、マーク、クリップ、クリップシーケンスの削除方法について説明する。図24の録画・編集・再生制御部がマーク削除要求を受けた場合の処理について例を用いて説明する。削除は、前述したマーク、クリップ、クリップシーケンスの追加と丁度正反対になる。即ち、前述した追加の例で使用した図39がAVクリップ管理情報の削除前の図であり、図38がAVクリップ管理情報を削除した後の図になる。

【0244】まず、クリップシーケンスの削除方法について説明する。削除するクリップシーケンス(Clip Sequence)をエントリークリップ(Entry Clip)の情報ごとAV Clip Sequenceから削除する。同時に、管理しているクリップシーケンス数(Num of Clip Sequences)を削除したクリップシーケンスの分だけ削減する。

【0245】次に、クリップの削除方法について説明する。削除するクリップ(Clip)をクリップテーブル(Clip Table)から削除し、管理しているクリップ数(Num of Clips)を削除したクリップの分だけ削減する。また、削除したクリップを含むクリップシーケンスがある場合は、このクリップシーケンスは削除される。

【0246】次に、マークの削除方法について説明する。削除するマーク(Mark)をマークテーブル(Mark Table)から削除し、管理しているマーク数(Num of Marks)を削除したマークの分だけ削減する。また、削除したマークを含むクリップがある場合は、このクリップは削除され、削除されたクリップを含むクリップシーケンスも削除される。

【0247】(AVクリップ管理情報のSplit)編集時のSplitに伴なうAVクリップ管理情報の更新について説明する。

【0248】図24の特再情報編集部123が、Split要求を受けた時の動作について例を用いて説明する。

【0249】図40は、Split前のAVクリップ管理情報を示し、図41は、このAVクリップ管理情報をフレーム00:00:02:00でSplitした後の2つのAVクリップ管理情報を示す。

【0250】尚、Splitするフレーム00:00:02:00がどちらのAVクリップに属するかは本発明の本質と関係がないので、本例では後半部のAVクリップに属するものとして以下説明をするが、Splitするフレームの扱いが制限されるものではない。

【0251】まず、前半部のAVクリップ管理情報の内容について説明する。前半部のAVファイルが、もとのAVファイルの内容を引き継ぐため、AV\_File\_IDはSplit前と同じである。もとのAVファイルに比べて失った区間のマーク(図40のMark#2)と、このマークを含むクリップ(図40のClip#1)が削除される。代わりに新たなAVファイルのビデオ表示終了フレームを示す新マーク(図41のMark#2)が作られ、AVファイルのビデ

オ表示開始フレームを示すマーク(図41(a)のMark

#1) をStart Mark、AVファイルのビデオ表示終了フレームを示す新マーク(図4 1 (a) のMark#2)をEnd Markとする新たなクリップ(図4 1 (a) のClip#1)が作成される。また、VOB後半を失うため、VOBのビデオ表示終了時刻VOB\_V\_E\_PT<sub>M</sub>は、新しいマーク(図4 1

(a) のMark#2)の表示終了時刻となる"270270"に変更され、VOB最後のパックのバッファへの入力時刻LAST\_SCRも実際のストリーム中の値に変更される(本例においては"160270")。

【0 2 5 2】次に、後半部のAVクリップ管理情報の内容について説明する。後半部のAVファイルのAV\_File\_IDには新たな名前が記録される。もとのAVファイルに比べて失った区間のマーク(図4 0のMark#1)と、このマークを含むクリップ(図4 0のClip#1)が削除される。代わりに新たなAVファイルのビデオ表示開始フレームに新マーク(図4 1 (b) のMark#1)が作られる。また、前半部を失うため残りのマーク(図4 0のMark#2)は失った時間00:00:02:00だけ早まった時刻で記録される。また、AVファイルのビデオ表示開始フレームのマーク(図4 1 (b) のMark#1)をStart Mark、AVファイルのビデオ表示終了フレームのマーク(図4 1

(b) のMark#2)をEnd Markとする新たなAVクリップ(図4 1 (b) のClip#1)が作成される。また、VOB前半を失うため、VOBのビデオ表示開始時刻VOB\_V\_S\_PT<sub>M</sub>は、新しいマーク(図4 1 (b) のMark#1)の表示開始時刻となる"270270"に変更され、VOB先頭パックのバッファへの入力時刻を示すFIRST\_SCRも実際のストリーム中の値に変更される(本例においては"170270")。

【0 2 5 3】次に、クリップシーケンスの情報であるAV\_Clip\_Sequenceについて説明する。Splitにより失われたクリップ(図4 0のClip#1)を含むクリップシーケンスは削除される。代わりに新たに作成されたクリップ

(前半部のAVファイルでは図4 0 (a) のClip#1、後半部のAVファイルでは図4 1 (b) のClip#1)を再生するための新たなクリップシーケンス(図4 1 (c) のClip\_Sequence#1およびClip\_Sequence#2)が作成される。

【0 2 5 4】尚、縮集時のShortenの場合は、Splitと、SplitによってできるAVクリップ管理情報の一方を削除する動作と等価であるため、本例での説明は省略する。

【0 2 5 5】(AVクリップ管理情報のMerge)縮集時のMergeに伴なうAVクリップ管理情報の更新について説明する。

【0 2 5 6】図2 4の特再情報縮集部がMerge要求を受けたときの動作について例を用いて説明する。

【0 2 5 7】図4 2は、Merge前の2つのAVクリップ管理情報を示す。図4 3は、この2つのAVクリップ管理情報の間に再エンコードした2つのVOBUを追加してMergeした後のAVクリップ管理情報を示す。前方のAVファ

イルの後には12フレームからなるVOBUを、後方のAVファイルの前には18フレームからなるVOBUを追加する。また、2つのVOBUは夫々接続するVOB(AVファイル)のタイムスタンプに連続してエンコードされたデータであるものとする。

【0 2 5 8】MergeされたAVクリップ管理情報の内容について説明する。前方のAVファイルをMerge後のAVファイルが引き継ぐため、AV\_File\_IDには前方のAVファイルの名前が記録される。前方のAVファイルには12フレームのVOBUが追加されるので、AVファイルのビデオ表示終了フレームを示すマーク(図4 2 (a) のMark#2)が12フレーム増加する。同様に後方のAVファイルには18フレームのVOBUが追加されるので、AVファイルのビデオ表示終了フレームを示すマーク(図4 2 (b) のMark#2)が18フレーム増加する。

【0 2 5 9】2つのAVファイルはMergeによって1つのAVファイルになるが、お互いの時間属性には相関が無いため、2つのAVファイル間でタイムスタンプの不連続が生じる。このため、Mergeされても、異なるVOBとして夫々が管理される。前方のAVファイルのVOB\_IDを1、後方のAVファイルのVOB\_IDを2としてAVクリップ管理情報に記録する。

【0 2 6 0】また、前述したオーディオギャップで説明したように、ビデオとオーディオのフレーム周期の違いから、VOB間でオーディオギャップが発生する。オーディオギャップで述べたように、VOB先頭のビデオ表示開始時刻を基準にオーディオのギャップ開始時刻およびギャップ時間を求めると、オーディオギャップ開始時刻A\_STP\_PT<sub>M</sub>=216810、オーディオギャップ時間長A\_GAP\_LEN=378となる。

【0 2 6 1】マークおよびクリップは、VOB内で完結するので、先に説明したMergeで追加されるVOBUの分だけが変更されるが、前方のVOBには後ろに12フレーム、後方のVOBには前に18フレーム夫々増加するので、前方VOBのビデオ表示終了時刻VOB\_V\_E\_PT<sub>M</sub>は12フレーム分だけ増加し、後方のVOBのビデオ表示開始時刻VOB\_V\_S\_PT<sub>M</sub>は18フレーム分だけ削減される。

【0 2 6 2】また、増加したフレーム分のパックが夫々に増えているので、前方のVOBのLAST\_SCRは、増加したパックの最後のSCRに変更され、後方のVOBのFIRST\_SCRは、増加したパックの先頭のSCRに変更される。また、マージされた2つのVOB間で、シームレス再生(プレゼンテーションデータ参照)が可能な場合は、後部VOBのSeamless\_Flagを"YES"に変更する。

【0 2 6 3】次に、AVクリップシーケンスの情報であるAV\_Clip\_Sequenceについて説明する。

【0 2 6 4】AVクリップシーケンスを構成するEntry Clipは基本的に変更されないが、AVファイルがMergeされることによる各Entry Clipの属性が変更される。Entry Clip#1は前方のAVファイル中に含まれているの

で、属性情報はそのまま継承される。Entry Clip#2は後方のAVファイル中に含まれているので、属性情報であるAV File IDは"System\_2"に、VOB IDは"2"に変更される。また、Mergeによって2つのVOB間がシームレス再生可能になった場合は、シームレス再生される後方VOBを示すEntry Clip中のSeamless Flagが"YES"に変更される。このSeamless Flagの判定は次のようにして行う。

【0265】Entry Clip間でのシームレス再生の判定手順について図44を用いて説明する。

【0266】まず、Entry Clip間でAV File IDが一致するかを調べる。一致する場合はステップ2へ移り、一致しない場合はSeamless Flagを"NONE"にする(ステップ1)。

【0267】次に、Entry Clip間でVOB IDが一致するかを調べる。一致する場合はステップ3へ移り、一致しない場合はステップ4に移る(ステップ2)。

【0268】次に、ステップ2でEntry Clip間でVOB IDが一致する場合について説明する。AVファイル内でEntry Clipが連続しているかを調べ、連続する場合はSeamlessFlagを"YES"にし、連続しない場合はSeamless Flagを"NONE"にする(ステップ3)。

【0269】次に、ステップ2でEntry Clip間でVOB IDが一致しない場合について説明する。AVファイル内でVOBが連続しているかを調べ、連続する場合はステップ5に移り、連続しない場合はSeamless Flagを"NONE"にする(ステップ4)。

【0270】次に、前Entry ClipがVOB終端のClip、且つ、後Entry ClipがVOB先端のClipであるかを調べる。本条件を満たす場合はSeamless Flagを"YES"にし、満たさない場合は、Seamless Flagを"NONE"にする(ステップ5)。

【0271】また、Mergeで、2つのAVファイルに1つのVOBUを追加するのは上述した例で片方のVOBUのサイズが0であるものと同一であり、2つのAVファイルを直接マージするのは上述した例で両方のVOBUのサイズが0であるものと同一であり、1つのAVファイルに1つのVOBUを追加するのは上述した例で片方のファイルが無く、この空のファイルに追加するVOBUのサイズが0であるものと同一である。また、上述した例では1つのファイルに追加するVOBUを1つであるとしたが、VOBUの数は本質的に1つに限定されるものではない。

【0272】以上でディスク上に記録されるファイルの論理フォーマットの説明を終わり、以下DVDレコーダーの動作について図24の構成図を基に説明する。

【0273】6. DVDレコーダーの動作：各動作にU1の説明、ファイル操作を付加

#### 6. 1. 初期状態

DVDレコーダーに新しいディスクが装着されると、ディスク読み出し部101はディスクの状態を調査するためにディスクの回転が安定するのを待った後、ファイルシス

テム部102に対してDVD-RAMメディアが確かに装着され安定読み出し状態になった事を通知する。

【0274】ファイルシステム部102はディスク読み出し部101に対してボリューム情報とファイル情報、AVクリップ管理情報の読み出しを指示する。読み出されたボリューム情報とファイル情報、AVクリップ管理情報はファイルシステム部102のメモリに記憶され、以降のファイル操作に対してはこのメモリ中のデータを参照する。これらの情報はディスクの取り出しを指示されるまで、ファイルシステム部102内のメモリに保持される。

【0275】録画・編集・再生制御部105は、ユーザIF部106に対して、読み出されたAVクリップ管理情報からディスク内に記録されているAVファイルの状態をグラフィックスし、図46に示す初期画面として表示することを要求する。ディスク内に何も記録されてない場合はカーソルの矢印のみが表示される。

【0276】図46の初期画面において、ユーザーが図45に示すリモコンのボタンを操作することにより、再生、編集、録画のそれぞれの状態に移行する。

#### 【0277】6. 2. 録画

図46の初期画面で、ユーザーが図45に示すリモコンの録画ボタンを押すと、図64の201の画面へと移行してくる。本実施例では設定可能な録画条件は録画時間と録画品質の2点とする。

【0278】まず「1」「選択」を押すと録画時間の指定にフォーカスがあたり、図45のリモコンボタンのカーソルボタンにより「無制限」または「指定時間」に矩形を移動した後、再度「選択」ボタンを押すことにより条件が設定される。

【0279】「指定時間」が選択された場合には、テンキーボタンにより時間を入力する画面に切り替わり、設定が完了すると再度図64の201の画面に戻る。録画条件の「録画品質」はAVデータのビットレートや解像度を制御し、図24のAVデータ入力部111で発生するMPEGデータのビットレートが制御される。

【0280】高画質、標準、時間優先に対する制御方法を図65に示す。図64の202に移行した後、リモコンの「録画」ボタンが押されたとする。

【0281】録画処理を開始する前に、図24の録画・編集・再生制御部105は102のファイルシステム部に対して図1のコマンドの中から「CREATE」を要求し、ファイルシステム部はファイルが作成できる場合にはファイル識別子を返す。この際、録画条件に合わせてファイルサイズはディスクの最大サイズが取られる。録画・編集・再生制御部105からAVデータ録画部110にファイル識別子と録画条件である図65の時間優先のパラメータが渡され録画開始が指示される。

【0282】AVデータ録画部110では内部のバッファメモリであるトランクバッファ1.5MBの内容をクリア

し、AVデータの入力が可能となる。

【0283】AVデータ入力部111では入力されるAV信号のエンコードが開始される。エンコードされた結果発生するビット列は絶えずトラックバッファに記録される。特殊再生情報生成部112ではAVデータ入力部がAVデータに挿入するGOP境界識別子及びGOP先頭のI-pictureの境界の検出を行い、検出されるまでにカウントされたビット量をセクター数に換算し内部のメモリに記憶する。

【0284】AVクリップ管理情報生成部114では、AVデータの記録開始時にメモリ中にAVクリップ管理情報のデータ領域を新しく作り、ビデオデータ開始地点をマークしメモリに記憶する。この時、他の領域は初期値（もしくは未定義）のままである。

【0285】トラックバッファがあらかじめ定められた容量まで占有された時点ではオーバーフロー対策部113がトラックバッファの内容を読み出しAVファイルシステム部103に対して図1の「AV-WRITE」コマンドを用いてディスクへの書き込みを要求する。AVデータ入力部111が発生するビットレートよりファイルシステムを通してディスクに記録されるビットレートが高いため、図5のバッファリングモデルに示されるようにトラックバッファ内のデータ占有量は減少する。このAVデータ録画部における処理と並行してAVファイルシステム部103を通じて記録要求されたAVデータはディスク記録部100によりディスクに記録されていく。この際の記録方法を図7を用いて説明する。

【0286】図7は図4でフォーマットを説明したAVブロック管理テーブルの現在の状態を示すものである。AVブロック管理テーブルのアドレス（上位）0、（下位）0,1,2はボリューム情報、ファイル管理情報、AVファイル管理情報ファイル、AVクリップ管理情報ファイルなど非AVファイル（情報）用にあらかじめ使用することが示されている。これらの情報は実際のディスク上にはまだ記録されていない。一方記録を開始したAVデータはアドレス（上位）1、（下位）0のAVブロックから記録を開始する。これは図7中の斜線で示されたブロックである。ディスクの初期状態から記録を開始するので、順次連続するAVブロックに連続的に記録がなされた後、ゾーン1とゾーン2の境界でゾーンを2に移り、以降ゾーン2、ゾーン3とゾーン内では連続的に記録が行われ対応するAVブロック管理テーブルの内容も変化する。ここでAVブロック管理テーブル自身も図24のファイルシステム部102のメモリ中に記憶されているためディスクにはAVデータのみが書かれるため無駄なシークや回転待ちは一切発生しない。

【0287】図8にはファイルシステム情報の中のファイル情報の状態を示す。DVD-RAMのファイルシステムにおいてファイルは複数のExtentのリンクリストとして管理される。一つのExtentはディスク上における連続記録

領域に対応する。先に説明した様にゾーン境界にはセクタバッファが存在するため、ゾーンの境界では連続記録ができず新たなExtentが生成される。

【0288】ゾーン境界では数100msの遅延が生じる。しかしAVブロック内の全てのセクターは連続記録を行っているので図24のAVデータ録画部110中のトラックバッファがオーバーフローを起こすことはなく、連続的なデータ入力が行われる。

【0289】このようにAVデータの録画を行った結果、ディスクにもはやバッファリングされている以上のデータが記録できなくなった場合の動作を説明する。図24中のAVデータ録画部において、AVデータ入力部111はエンコード処理を中止する。MPEGにおいて有効なデータ境界は必ずGOP境界でなければならないため、中断される直前のGOP境界がデータの最終ポイントとなる。

【0290】特殊再生情報生成部112では録画を開始後、全てのGOP境界とGOP先頭のI-picture境界を検出、各GOPのセクター数及びGOP先頭のI-pictureのセクター数を図58に示す配列の形態で保持している。これらの情報は制御データ管理部107に転送されAVファイル管理情報ファイルの生成に用いられる。またオーバーフロー対策部113は最後のGOP境界までのAVデータを有効データとしてAVファイルシステム部103に転送を行い、AVデータ録画部110における全ての録画処理が終了する。

【0291】AVクリップ管理情報生成部114は、録画終了時にAVデータの終了時点をマークし、これをAVクリップ管理情報として、メモリに書き込む。このAVクリップ管理情報は、制御データ管理部107に転送され、AVクリップ管理情報ファイルAV Clip part, AV Clip Sequenceの生成や更新に用いられる。

【0292】制御データ管理部107は引き渡された図58に示される特殊再生制御情報や図37のAVクリップ管理情報に基づきAVファイル管理情報ファイルとAVクリップ管理情報ファイルを非AVファイルシステム部104を通じてディスク上に記録する。ここで、AVクリップ管理情報内に記録された録画されたAVデータの開始点から終了点までを1クリップ（マーク位置は開始点と終了点）とし、クリップ1つのAVクリップ列が自動的に生成され、AVクリップ管理情報AV Clip Sequenceファイルを新規ディスクであれば作成し、他のAVデータが記録済みのディスクであれば、新たなクリップをAVクリップ管理情報AV Clip Sequenceに追加更新する。この時、この新しいクリップにはSeamless InformationとしてNon Seamlessがセットされる。

【0293】また、このディスク記録に先立ち、ファイルシステム部102に記憶されていたボリューム情報や図8に示されるExtentなどのファイル情報もディスク上に記録される。これらの情報の記録を終えた後もボリュ

ーム情報やファイル情報はファイルシステム内のメモリ中に残り、次の録画や編集に使用される。

【0294】初期状態以外の状態で録画を行う場合には、図45のリモコンの録画ボタンを押すと録画の初期画面が現れ、上記と同じ手順で録画を行うことが出来る。

### 【0295】6. 3. 仮編集

次に、編集の手順について説明する。

【0296】本実施例における編集は、AVクリップによる仮編集と、実際にMPEGストリームへの加工（再編集）を行う本編集とから構成される階層式編集である。ここでは、仮編集について説明する。

【0297】（マーク設定）まず、AVファイルにマークを設定する手順について説明する。

【0298】図46で示す初期状態から一つのクリップシーケンスを選択して再生を行う。再生の詳細については後述する。

【0299】図47は、マーク設定の流れを示した図である。ユーザは再生画面を見ながら気に入った画面でリモコンのマークボタンを押す。この操作によって、リモコンからユーザIF部を通して、録画・編集・再生制御部へマーク要求が送られる。録画・編集・再生制御部はAVデータ再生部へ、再生中のAVファイルの識別子であるAV File IDと、タイムスタンプ（AVファイル内の相対値）を要求し、AVデータ再生部は、録画・編集・再生制御部にAVFile IDおよびタイムスタンプを返す。録画・編集・再生制御部は、AVデータ編集部に対して、再生中のAVファイルIDとタイムスタンプを添えてマーク要求を送る。AVデータ編集部のAVクリップ列編集部では、AVクリップ管理情報中のAV Clip partの情報を用いて、受け取ったAVファイルIDとタイムスタンプから再生中のVOBを特定し、このVOBのビデオ表示開始時刻VOB\_V\_S\_PTMと受け取ったタイムスタンプを比較し、マークが指定されたフレームのVOB内での相対タイムコードを特定する。AVクリップ列編集部は、要求されたマーク情報を、”AVクリップ管理情報の作成”のマーク追加手順に従い、AVクリップ管理情報に追加する。このAVクリップ管理情報は、AVデータの再生終了後に、制御データ管理部に送られ、非AVファイルシステム部を通してディスクに記録される。

【0300】次に、録画時のマーク設定手順について説明する。図46で示す初期状態から、前述した録画の手順に従い、録画を行う。

【0301】図48は、録画時のマーク設定の流れを示した図である。ユーザは録画画面を見ながら気に入った画面でリモコンのマークボタンを押す。この操作によって、リモコンからユーザIF部を通して、録画・編集・再生制御部へマーク要求が送られる。録画・編集・再生制御部はAVデータ録画部へ、録画中のAVファイルの識別子であるAV File IDと、タイムスタンプ（AVファイル

ル内の相対値）を要求し、AVデータ録画部は、録画・編集・再生制御部にAVFile IDおよびタイムスタンプを返す。録画・編集・再生制御部は、AV File ID、仮マークIDとタイムスタンプから構成される仮マークテーブルを作成し、この仮マークテーブルへ仮マークおよびタイムスタンプの登録を行う。以降、マーク要求がある度に、同様の処理を行い、仮マークテーブルへの登録を行う。録画・編集・再生制御部は、録画終了と同時に、仮マークテーブルに登録された仮マーク毎に、AVデータ編集部に対して、AVファイルIDとタイムスタンプを添えてマーク要求を送る。以後の処理は、前述した再生時のマーク設定方法と同様である。

【0302】次に、録画時の自動マーク設定方法について説明する。AVデータ録画部で、シーンチェンジを検出した場合、AVデータ録画部は、自動的に録画・編集・再生制御部にAV File IDと、タイムスタンプを送る。録画・編集・再生制御部では、AVデータ録画部より送られたAV File IDと、タイムスタンプを前述した仮マークテーブルに自動的に登録を行う。以後、前述した録画時のマーク設定手順と同様の処理を行い、マークを設定する。

【0303】次に、再生時の自動マーク設定方法について説明する。AVデータ再生部で、シーンチェンジを検出した場合、AVデータ再生部は、自動的に録画・編集・再生制御部にAV File IDと、タイムスタンプを送る。録画・編集・再生制御部では、AVデータ再生部より送られたAV File IDと、タイムスタンプをマーク要求と共にAVデータ編集部に送る。以後、前述した再生時のマーク設定手順と同様の処理を行い、マークを設定する。

【0304】（クリップ設定）次に、AVファイルにクリップを設定する手順について説明する。

【0305】図49は、クリップ設定の流れを示した図である。図46で示す初期状態で、リモコンの仮編集ボタンを押すと、録画・編集・再生制御部は、制御データ管理部よりAVクリップ管理情報を読み出し、ユーザIF部を通して、クリップ設定の画面を表示する。具体的には、図49に示すような、録画の記録であるAVデータの録画日、録画時間、録画したチャンネルと、録画したデータを時間軸で示す簡単なグラフと、AVデータに登録されているマークおよびクリップと、クリップシーケンスと、一つのマークまたはクリップを選択するカーソルが表示される。ユーザはリモコンの上下左右キーを押し、フォーカスされるマークまたはクリップを移動する。ユーザは、選択キーを使い2つのマークをリモコンで選択する。この操作によって、2つのマークに挟まれた区間が1つのクリップとなるように、ユーザIF部を通して、録画・編集・再生制御部にクリップ要求が送られる。録画・編集・再生制御部は、AVデータ編集部に、選択された2つのマークと、2つのマークが属するAVファイルIDおよびVOB IDを添えてクリップ要求を送る。

AVデータ編集部のAVクリップ列編集部は、要求されたクリップ情報を、"AVクリップ管理情報の作成"のクリップ追加手順に従い、AVクリップ管理情報に追加する。このAVクリップ管理情報は、制御データ管理部に送られ、非AVファイルシステム部を通してディスクに記録される。

【0306】また、カーソルの移動に応じてカーソルが指し示すマークのフレームやマーク以降の映像をバックグラウンドに表示することも可能である。詳しくは、クリップシーケンスの再生で説明する再生手順と同様である。

【0307】(クリップシーケンス設定)次に、クリップシーケンスを設定する手順について説明する。

【0308】図50は、クリップシーケンス設定の流れを示した図である。クリップ設定と同じく、図46で示す初期状態で、リモコンの仮編集ボタンを押すと、録画・編集・再生制御部は、制御データ管理部よりAVクリップ管理情報を読み出し、ユーザIF部を通して、クリップシーケンス設定の画面を表示する。具体的には、図50に示すような、録画の記録であるAVデータの録画日、録画時間、録画したチャンネルと、登録されているマークと、クリップと、一つのマークまたはクリップを選択するカーソルと、クリップシーケンスが表示される。ユーザはリモコンの上下左右キーを押し、フォーカスされるマークまたはクリップを移動する。ユーザは、選択キーを使い複数のクリップをクリップの再生順に従い、リモコンで選択する。この一連の操作によって、選択されたクリップ列が一つのクリップシーケンスとなるように、ユーザIF部を通して、録画・編集・再生制御部にクリップシーケンス要求が送られる。録画・編集・再生制御部は、AVデータ編集部に、選択されたクリップ列と、各クリップが属するAVファイルIDとVOB IDとを添えてクリップシーケンス要求を送る。AVデータ編集部のAVクリップ列編集部は、要求されたクリップシーケンス情報を、"AVクリップ管理情報の作成"のクリップシーケンス追加手順に従い、AVクリップ管理情報に追加する。このAVクリップ管理情報は、制御データ管理部に送られ、非AVファイルシステム部を通してディスクに記録される。

【0309】また、カーソルの移動に応じてカーソルが指し示すクリップをバックグラウンドに表示することも可能である。詳しくは、クリップシーケンスの再生で説明する再生手順と同様である。

【0310】(クリップシーケンス削除)次に、マーク、クリップ、クリップシーケンスの削除手順について説明する。

【0311】図51は、マーク、クリップ、クリップシーケンス削除の流れを示した図である。

【0312】クリップ、クリップシーケンスの設定と同じく、図51で示す初期状態で、リモコンの仮編集ボタ

ンを押すと、録画・編集・再生制御部は、制御データ管理部よりAVクリップ管理情報を読み出し、ユーザIF部を通して、マーク、クリップ、クリップシーケンス削除の画面を表示する。具体的には、図51に示すような、録画の記録であるAVデータの録画日、録画時間、録画したチャンネルと、登録されているマークと、クリップと、クリップシーケンスと、マークまたはクリップまたはクリップシーケンスを選択するカーソルが表示される。ユーザは、リモコンの上下左右キーを押し、カーソルを削除したいマークまたはクリップまたはクリップシーケンスに移動し、削除キーを押す。この操作によって、選択されたマークまたはクリップまたはクリップシーケンスを削除するように、ユーザIF部を通して録画・編集・再生制御部に削除要求が送られる。録画・編集・再生制御部は、AVデータ編集部に、選択されたマークまたはクリップまたはクリップシーケンスと、マークまたはクリップの場合は各マークまたはクリップが属するAVファイルIDとVOB IDとを添えて削除要求を送る。AVデータ編集部はのAVクリップ列編集部は、要求された削除情報を、"AVクリップ管理情報の削除"の手順に従い、AVクリップ管理情報から削除する。このAVクリップ管理情報は、制御データ部に送られ、非AVファイルシステム部を通してディスクに記録される。

【0313】6. 4. AVファイル編集(本編集)

次に、AVファイル編集(本編集)について説明する。AVファイル編集(本編集)は、前述した仮編集で作成した複数のクリップシーケンスの中から1つのクリップシーケンスを選択し、実際にMPEGストリームに対して加工(編集)を行うものである。

【0314】(AVファイル編集の設定)次に、クリップシーケンスにより論理的に仮編集されたAVデータに対しての実際の編集を設定する手順について説明する。

【0315】図52は、編集の設定の流れを示した図である。図46で示す初期状態で、リモコンの本編集ボタンを押すと、録画・編集・再生制御部は、制御データ管理部よりAVクリップ管理情報を読み出し、ユーザIF部を通して、編集を行うクリップシーケンス設定の画面を表示する。具体的には、図52に示すような、録画の記録であるAVデータの録画日、録画時間、録画したチャンネルと、マークと、クリップと、クリップシーケンスと、一つのクリップシーケンスを選択するカーソルが表示される。ユーザはリモコンの上下左右キーを押し、フォーカスされるクリップシーケンス移動する。ユーザは、選択キーを使い複数のクリップシーケンスの中から、気に入った一つのクリップシーケンスを選択する。この操作によって、選択されたクリップシーケンスに対して実際の編集が行われるように、ユーザIF部を通して、録画・編集・再生制御部に編集要求が送られる。録画・編集・再生制御部は、AVデータ編集部に、選択されたクリップシーケンスを示すIDを添えて編集要求を送

る。AVデータ編集部のAVクリップ編集部は、要求された編集情報を、後述するAVファイル、AVファイル管理情報およびAVクリップ管理情報の編集手順に従い編集を行う。編集されたAVファイルは、AVファイルシステム部を通してディスクに記録され、AVファイル管理情報およびAVクリップ管理情報は、制御データ管理部に送られ、非AVファイルシステム部を通してディスクに記録される。

【0316】(AVファイル編集の動作)次に、AVファイル、AVファイル管理情報およびAVクリップ管理情報の編集手順について説明する。

【0317】図53は、編集の流れを示した図である。図53は、図52で示すクリップシーケンスの内、クリップシーケンス#3が選択され、編集される場合の例を示した図である。尚、本発明のAVファイル編集とは、クリップの取り出しと、クリップの接続を効率良く組み合わせることで実現するものである。

【0318】まず最初に、AVファイルから編集で使用する各クリップを取り出す。AVファイル"System\_1"には、使用するクリップが2つ存在するので、夫々を取り出せるように、AVファイルシステムで用意されているSplit機能を使いAVファイル"System\_1"を2つのAVファイルに分割する。同様に、AVファイル管理情報、AVクリップ管理情報も、夫々用意しているSplit機能を使い2つのAVファイルに分割する(ステップ1)。

【0319】次に、各クリップの両端であるスタートマークおよびエンドマークを含むAVデータを読み出す。多重化の説明でも述べたが、MPEGの性質上ビデオとオーディオのデータは最大1秒離れて多重化が行われるため、各クリップ両端のビデオおよびオーディオデータ両方を読み出すためには、マークが指すビデオフレームを含むVOBUだけでなく、その前後の1秒分のビデオデータを含む一つまたは複数のVOBU(以降「VOBU群」と呼ぶ)を読み出すことが必要である。まず、指定されたクリップの属するAVファイルをオープンする。次にスタートマークのタイムコードとAVファイル管理情報中のVOBU Map中のStart Timeとの比較を行い、スタートマークの指すビデオフレームの前後1秒分のビデオデータを含むVOBU群を特定する。次に、VOBU Map中のStart Sector情報を用いて読み出すVOBU群の開始セクタと終了セクタを特定し、VOBU群開始セクタまでAVファイルのSEEKを行い、VOBU群終了セクタまでAVファイルのREADを行う。エンドマークも同様の処理を行い、VOBU群の読み出しを行う。以上のようにして読み出したVOBU群に対して、VOBU間がシームレス再生可能となるかを読み出したVOBU群の解析により決定し、再エンコード方法の説明で述べた手順に従い、ビデオ再エンコードと、オーディオギャップの作成(シームレス再生時のみ)またはオーディオデータの切り出し(非シームレス再生時のみ)と、ビデオ、オーディオデータの多重化を行う。尚、読み出すVOBU群

10

20

30

40

50

の決定方法は、上記に限定されるものではなく、AVデータの解析を行い、必要最小限のAVデータのみを読み出しても良いし、逆に多くても(例えば10VOBU固定のようにする)構わない(ステップ2)。

【0320】次に、各AVファイルから不要部分を取り除き、クリップだけのAVファイルを作る。ステップ2でクリップ切断面を含むVOBU群を読み出し、再エンコードおよび多重化を行っているので、この読みだしを行ったVOBU群を含んで不要データとして切り捨てる。AVファイルからのデータの削除は、AVファイルシステムで用意されているShorten機能を使用する。同様に、AVファイル管理情報、AVクリップ管理情報も夫々用意されているShorten機能を使用して管理情報を変更する(ステップ3)。

【0321】最後に、クリップを再生シーケンスの順番で接続していき、一つのAVファイルへとマージを行う。まず、最初のクリップ(図53のクリップ(3))と二番目のクリップ(図53のクリップ(4))を1つのAVファイルへとマージを行う。この時、ステップ2で読み込み、再エンコードを行ったVOBU群<32'>(ステップ2で読み込んだVOBU群<32>を再エンコードしたVOBU群)およびVOBU群<41'>(ステップ2で読み込んだVOBU群<41>を再エンコードしたVOBU群)を間に挟み込む。2つのクリップと2つのVOBU群は、AVファイルシステムで用意されているMerge機能を使用することで、1つのAVファイルへとマージがされる。また、同様に、AVファイル管理情報、AVクリップ管理情報も、夫々用意されているMerge機能を使用して1つのAVファイル管理情報、AVクリップ管理情報にマージされる。同様に、マージされたAVファイルと、3つ目のクリップ(図53のクリップ(5))と、VOBU群<42'>(ステップ2で読み込まれたVOBU群<42>を再エンコードしたVOBU群)と、VOBU群<51'>(ステップ2で読み込まれたVOBU群<51>を再エンコードしたVOBU群)をマージする。最後に、先頭にVOBU群<31'>(ステップ2で読み込まれたVOBU群<31>を再エンコードしたVOBU群)と最後にVOBU群<52'>(ステップ2で読み込まれたVOBU群<52>を再エンコードしたVOBU群)を夫々Merge機能により接続する(ステップ4)。

【0322】以上の処理によって、論理的に仮編集されたクリップシーケンスを、実態の伴なう1つのAVファイルへと編集を行うことができる。また、本編集では、MPEGストリームのタイムスタンプの付け替えを行わないことと、光ディスク上に記録されているMPEGストリームの移動を必要最小限に抑えていることにより、従来のMPEGストリームの編集では得られなかった短時間(数秒)での編集が可能である。

【0323】次に、図54を用いて、本実施例における階層式編集方法について説明する。本実施例における階層式編集方法は、次の3つのステップから成り立っている。

- 【0324】1) 仮編集ステップ：マーク、クリップおよびクリップシーケンスによる編集  
2) 確認ステップ：仮編集結果の確認  
3) 本編集ステップ：MPEGストリームの加工（編集）

まず、”仮編集ステップ”について説明する。仮編集は大きく2つのステップに分かれる。前半はマーク設定ステップで、後半はクリップおよびクリップシーケンスの設定ステップである。まず、マークの設定は、MPEGストリームの再生中に、ユーザからのマーク要求を受けるマーク要求ステップと、マーク要求がある場合にマークを設定するマーク設定ステップと、再生終了を判定する再生終了判定ステップとから構成される。後半は、ユーザからのクリップ要求を受けるクリップ要求ステップと、クリップ要求がある場合にクリップを設定するクリップ設定ステップと、ユーザからのクリップシーケンス要求を受けるクリップシーケンス要求ステップと、クリップシーケンスの要求がある場合にクリップシーケンスを設定するクリップシーケンス設定ステップと、クリップ編集終了を判定するクリップシーケンス判定ステップとから構成される。

【0325】次に、”確認ステップ”について説明する。ユーザから確認（再生）を行うクリップシーケンスの選択を受けるクリップシーケンス選択ステップと、ユーザが選択したクリップシーケンスの再生を行う再生ステップとから構成される。

【0326】次に、”本編集ステップ”について説明する。ユーザから本編集を行うクリップシーケンスの選択を受けるクリップシーケンス選択ステップと、選択されたクリップシーケンスがシームレス再生が可能となるようにMPEGストリームの加工を行うMPEGストリーム加工（編集）ステップとから構成される。

【0327】本実施例では、DVD-RAMを単なるAVデータの記録媒体としたのではなく、従来にない階層式編集を実現するAVデータおよび仮編集データの記録媒体としていることが特徴である。また、編集における全ての工程を1枚のDVD-RAM上で実現していることも特徴である。

#### 【0328】6. 5. 再生

（クリップシーケンスの再生）次に上記のように編集されたAVデータファイルを再生する場合の動作を説明する。図24のユーザIF部106は、録画・編集・再生制御部105より得たAVクリップ管理情報を基づいて、図46に示された画面により、ディスクに存在する全部のクリップシーケンスを表示する。その画面の中で、カーソルは1つのクリップシーケンスを指している。ユーザがリモコンの「上矢印」ボタンまたは「下矢印」ボタンを操作したときに、ユーザIF部106はクリップシーケンスを指しているカーソルを移動する。ユーザがリモコンで「再生」ボタンを押したとき、再生の動作を開始

10

20

30

40

50

する。なお、再生を中止する場合はリモコンの「STOP」ボタンを押す。

【0329】ユーザがリモコンで「再生」ボタンを押したとき、ユーザIF部106は録画・編集・再生制御部105に引数としてクリップシーケンスの番号を渡して再生処理を要求する。録画・編集・再生制御部105は再生を指示されると、引数として渡されたクリップシーケンスをAVデータ再生部130に転送し、再生を要求する。

【0330】AVデータ再生部130が、引数として渡されたクリップシーケンスの再生を行う場合の動作を説明する。

【0331】まず、図24のAVクリップ列再生部132は制御データ管理部107からAVクリップ管理情報をもらい、該当する番号のクリップシーケンスを検索する。次に、AVクリップ管理情報を利用して、クリップシーケンスの各クリップの先端と終端のマークに関する情報を作成し、その情報を引数として渡して、AVファイル読み出し部133に再生を要求する。同時に、各クリップ間のシームレス情報を引数として渡して、AVデコーダ部に再生を要求する。

【0332】まず、AVファイル読み出し部133は、制御データ管理部107に対して、AVファイル管理情報の転送を要求し、得られたAVファイル管理情報を参照して、引数で渡された各クリップの先端のマークに対応するVOBUの先頭のセクタアドレスと、終端のマークに対応するVOBUの終わりのセクタアドレスを求める。

【0333】ここで、マークの指定時刻からVOBUの先頭または終わりのセクタアドレスを求める動作について説明する。VOB InformationのStart Timeを参照して、指定時刻を含むVOBを探す。得られたVOBのTime Map Tableを参照する。 $((Time\ Unit) * (Time\ Map\ Index) + (Time\ Base))$ が各Time Mapの指す時刻を表しているため、マークの指定時刻の前後を指すTime Mapのインデックスを求める。そのインデックスをiと仮定する。Time Map#iが指すVOBUをVOBU1、Time Map#(i+1)が指すVOBUをVOBU2と仮定する。求めたいVOBUは、VOBU1からVOBU2までの間に存在する。Time MapのStart Timeを補正することにより、VOBU1とVOBU2の先頭の時刻を得ることができる。Time MapのVOBU Map IndexとVOBU Map Tableを参照することにより、VOBU1とVOBU2の間に存在するVOBUを得ることができ、VOBU MapのStart Timeを補正することにより、それらのVOBUの先頭時刻を得ることができる。以上で得られた各VOBUの先頭の時刻とマークの時刻を比較することにより、マークの時刻を含むVOBUを得ることができる。そのVOBUのTime MapのStart SectorまたはVOBUMapのStart Sectorを補正することにより、そのVOBUのAVファイルの先頭からのセクタアドレスを得ることができる。VOBU Map Tableにおいて、得られたVOBUの次のVOBUの先頭のセクタアドレスを同様にして求めることによ

り、必要とするVOBUの終わりのアドレスを得ることができる。

【0334】次に、AVファイル読み出し部133は、ファイルシステム部102に対して、1番目のクリップのAV File IDで示されたAVファイルの「OPEN」を要求し、その後、そのクリップの先端のマークに対応するVOBUの先頭のセクタアドレスまで「SEEK」を要求する。それから、一度に32KB程度の大きさでデータを、AVファイルからAVデータ再生部130内のトラックバッファへ読み込むために、ファイルシステム部102に対して「READ」を要求する。そのクリップの終端のマークに対応するVOBUの終わりのセクタまで読み出すまで、「READ」を繰り返し要求する。ただし、トラックバッファがフルになったときは「READ」要求をしないで、トラックバッファに空きができるのを待ち、トラックバッファに空きができたら、「READ」の要求を再開する。そして、クリップに対応するデータの読み出し完了後に、トラックバッファにAVデータの切れ目を示すエンドマークを書き出す。

【0335】AVファイル読み出し部133は、2番目以降のクリップに対しても、順に同様の動作を繰り返し、全部のクリップに対応するデータを、トラックバッファに書き出し続ける。

【0336】AVデコーダ部131は、再生を要求されたときに、トラックバッファがフルになるまで、トラックバッファを監視しながら待つ。トラックバッファが、フルになったときに、デコード動作を開始する。トラックバッファ内にエンドマークが存在し、次のクリップとの接続がノンシームレス接続の場合は、エンドマークのデータまで、一旦、デコードを停止する。そして、再び、トラックバッファがフルになったときに、デコード動作を開始する。トラックバッファ内にエンドマークが存在し、次のクリップとの接続がシームレス接続の場合は、先に説明したシームレス再生の動作を行う。

【0337】次にこのファイル内の連続読み込みにおけるファイルシステムとディスク読み出し部の動作について説明する。読み出し対象のAVファイルが図9に示すように、AVブロックに記録されているとする。AV\_BLK#iとAV\_BLK#1を除いて全てのAVブロックには5.5MB以上のデータが記録されているものとする。

【0338】ファイルシステム部102が「READ」要求を受け取ると、ファイル識別子を元にメモリ中に常駐しているファイル管理情報から該当ファイルのExtent情報を検索する。AVファイルのExtent#1から先頭のAVブロックのアドレス情報を取り出し、該当するAVブロックからデータを読み込む。ファイルシステム部102からAVデータを受け取った後のAVデコーダ部131とAVデータ再生部130の動作は先に説明したとおりである。今AV\_BLK#iに5.5MB以上のデータが記録されていたとする。この場合、デコードはAV\_BLK#iを読み

込んでいる途中で開始される。ファイルシステム部102はデータを読み込んでいるセクタと、AV\_BLK#iの終端セクタを比較して、等しい場合は次のExtent#2へジャンプする。この時トラックバッファにはジャンプ中にAVデコード部131へデータを連続的に供給しつづけることが可能なサイズのデータが蓄積されている。これはAV\_BLK#i内のセクタに連続的にAVデータが記録されており、そのデータを連続的に読み出している間にトラックバッファ内にフルにデータが蓄積されてする事が可能なのである。トラックバッファ内にデータがフルにある場合、およそ1.5秒間AVデコーダ部131にデータを供給することが可能である。この1.5秒という値は民生機における最外周から最内周へのジャンプに要する時間であり、ワーストケースを保証することが可能なのである。Extent間でゾーンが異なる場合でも、ゾーン境界を跨ぐジャンプを行ってデータ読み出すために必要な時間は数100msであるので、データの連続性が十分保証される。

【0339】(早送り再生)クリップシーケンスの再生中に、早送りが指示された場合の動作を説明する。クリップシーケンスの再生中に、図45のリモコンにおいて「早送り」ボタンを押すと、図24のユーザIF部106から録画・編集・再生制御部105に早送り再生処理を要求し、録画・編集・再生制御部105は、AVクリップ列再生部132に早送り再生処理を要求する。

【0340】AVクリップ列再生部132は、AVデコーダ部131にデコードの停止を要求する。AVデコーダ部は、停止したときのクリップと時刻に関する情報を、AVクリップ列再生部132に返す。次に、AVクリップ列再生部132は、トラックバッファをクリアした後に、引数として停止したときのクリップと時刻に関する情報を、AVファイル読み出し部133に渡して、早送り再生処理を要求する。同時に、AVデコーダ部131に、早送り再生処理を要求する。

【0341】AVファイル読み出し部133が、早送り再生処理を要求されたときの動作を説明する。まず、引数として渡された停止したときの時刻より、5秒先の時刻を含むVOBUを求める。このVOBUを求める動作は、再生のときに説明したマークの時刻を含むVOBUを求める動作と同じである。AVファイル読み出し部133は、求めたVOBUの先頭のAVファイル内でのセクタアドレスとリファレンスピクチャのVOBU先頭からの相対エンドアドレスを得る。リファレンスピクチャのデータがAVブロックの不連続境界を跨っているか否かをファイルシステム部102に「IN\_AV\_BLK\_BOUND」を要求することにより、判定する。AVブロックの不連続境界を跨いでいる場合には、隣接する次のVOBUを求め、そのVOBUの先頭のAVファイル内でのセクタアドレスとリファレンスピクチャのVOBU先頭からの相対エンドアドレスを得る。次

に、AVファイル読み出し部133は、再生のときに、

クリップのデータを読み出したのと同様の動作により、AVファイルからVOBUのリファレンスピクチャのデータを読み出し、トラックバッファに書き出す。AVファイル読み出し部133は、データの書き出し完了後に、トラックバッファにAVデータの切れ目を示すエンドマークを書き出す。次に、さらに5秒ごとに先、すなわち、10秒先、15秒先、…のVOBUに対して、順に同様の動作を行う。指定時刻がクリップの終端を越えた場合は、次のクリップの先頭のVOBUを対象し、そのVOBUから、5秒ごとに先のVOBUに対して、順に同様の動作を行う。

【0342】なお、VOBUの大きさはビットレートが8Mbpsで1.2秒としても、1.2Mバイトであり、AVブロックの大きさより小さいため、隣接する2つのVOBUがどちらもAVブロック境界を跨ることはない。

【0343】上記の説明においては5秒先のVOBUを求めるが、この秒数を変えることにより、早送り再生の速度を変えることができる。

【0344】AVデコーダ部133が、早送り再生処理を要求されたときの動作を説明する。トラックバッファがフルになるか、または、エンドマークがトラックバッファに書き出されたら、デコード動作を開始する。そして、エンドマークのところでデコードを停止する。この動作を、繰り返す。

【0345】(早戻し再生) AVファイルの再生中に、早戻し再生が指示された場合の動作は、上記の早送り再生の説明における、5秒ごとに先のVOBUを求める動作を、5秒ごとに戻したVOBUを求めるような動作に変更する以外は、同じ動作をする。

【0346】(再生途中での停止) なお、再生途中で停止が行われた場合、再生中であったクリップを停止箇所で2つに分割し、分割した前部クリップの再生済みを示すフラグ(Played)を再生済みにする。

【0347】(簡易再生装置) 図37のAVクリップ管理情報の論理フォーマットを示す。

【0348】第1の実施例におけるAVクリップ管理情報と比較して、MarkにVOBUの先頭のアドレスとタイムコードを示す情報が追加されている。そのため、AVファイル管理情報のVOBU情報がなくてもクリップ再生が可能となるため、主記憶容量の小さい再生装置でも、主記憶にAVファイル管理情報を記憶することなく、クリップ再生が可能となる。

【0349】7. 第2の実施例

第2の実施例におけるAVファイル管理情報の論理フォーマットについて図69を参照しながら説明する。ストリーム情報に関しては、第1の実施例と同じであるため、説明を省略する。

【0350】Time Map Tableには、Time Map Groupの数、Time Map GroupのMain Time Mapが指す時間間隔を秒で示すMain Time Unit、Time Map GroupのSub Time M

ap間の時間間隔を秒で示すSub Time Unit、Main Time Map及びSub Time Mapが指す時間を補正するTime Base、Time Map Groupの配列へのポインタが記録される。

【0351】Main Time Mapには、VOBUの先頭のセクタ単位のアドレスを示すStart Sectorとタイムコードを示すStart Time Offsetが記録される。Start Time Offsetは、Main Time Mapが指すタイムコードからの相対値を示す。Sub Time Mapには、VOBUの先頭のセクタ単位のアドレスを示すStart Sector Offsetとタイムコードを示すStart Time Offsetが、記録される。Start Sector Offsetは、Main Time Mapで指されたVOBUの先頭からの相対値で示す。Start Time Offsetは、Main Time Mapが指すタイムコードからの相対値を示す。

【0352】Time Map TableのMain Time Unitは大きな秒単位の時間間隔を示し、Sub Time Unitは上記のMain Time Unitの区間を細かく区切る秒単位の時間間隔を示す。Time Map Groupは、Main Time Mapと複数のSub Time Mapからなる。Main Time Mapは、Main Time Unitで示された時間間隔に存在するVOBUの情報を示す。Sub Time Unitは、Main Time Unitで示された時間間隔の区間内のSub Time Unitで示された時間間隔に存在するVOBUの情報を示す。Main Time Unitをt秒、Sub Time Unitをs秒とすると、Sub Time Mapの個数pは、 $(t/s - 1)$ となる。

【0353】(AVファイル管理情報の大きさ)第1の実施例と同様の例では、Main Time MapのStart Sectorが4バイト、Start Time Offsetが1バイト、End Sector of the Reference Pictureが1バイトである。Sub Time MapのStart Sectorが2バイト、Start Time Offsetが1バイト、End Sector of the Reference Pictureが1バイトである。

【0354】例えば、再生時間が7時間のAVデータを、Time Map TableのMain Time Unitが5秒、Sub Time Unitを0.5秒ととして記録するならば、Time Map Groupの大きさは42バイト、Time Map Tableの大きさは、約210キロバイトであるので、AVファイル管理情報の大きさも、約210キロバイトである。

【0355】特殊再生を行う場合に、Sub Time Mapを参照しないで、Main Time Mapのみを参照しても、特殊再生をすることが可能である。ただし、特殊再生の速度の種類が少なくなる。このような特殊再生の方法により、主記憶の容量が少ない再生装置でもVOBU Map Tableを主記憶に記憶しないで、特殊再生を実行することができる。さらに、Sub Time Mapを間引いて持つことにより、主記憶に記憶する特殊再生情報の大きさを調整することも可能である。

【0356】

【発明の効果】請求項1または請求項2の本発明によれば、VOBU情報やタイムマップ情報の変更のための作業量や光ディスクへのアクセス回数を、AVファイルの大き

さに依存しない一定値で押さえられることができる。  
 【0357】また、請求項3の本発明によれば、請求項1または請求項2の光ディスクを編集する編集装置とすることができる。  
 【0358】また、請求項4の本発明によれば、請求項3記載の編集装置により編集された光ディスクを再生する再生装置とすることができます。

## 【図面の簡単な説明】

- 【図1】ファイルシステム部の機能一覧を示す図
- 【図2】ディスク全体の構成図
- 【図3】AVブロックの構成図
- 【図4】AVブロック管理テーブルの構成図
- 【図5】AVデータ書き込み時のバッファリングモデルを示す図
- 【図6】AVデータ読み出し時のバッファリングモデルを示す図
- 【図7】初期状態から録画を開始した直後のAVブロック管理テーブルの状態を示す図
- 【図8】初期状態から録画を開始した場合のファイル情報の状態を示す図
- 【図9】AVデータのAVブロックへの割り当てとシンケンシャル読み出しを示す図
- 【図10】セクタビットマップの構成図
- 【図11】AVブロック管理テーブルの構成図
- 【図12】AVブロック属性変化によるセクタビットマップ処理を示す図
- 【図13】AVファイルの「DELETE」コマンドの処理を示す図
- 【図14】非AVファイルの「DELETE」コマンドの処理を示す図
- 【図15】「SHORTEN」コマンドの処理を示す図
- 【図16】「SPLIT」コマンドにおけるAVブロックの扱いを示す図
- 【図17】「MERGE」コマンドにおけるAVブロックの扱いを示す図
- 【図18】「MERGE」コマンドにおけるAVブロックの扱いを示す図
- 【図19】「IN\_AV\_BLK\_BOUND」コマンドの処理を示す図
- 【図20】AVブロック不連続境界の説明図
- 【図21】「MERGE」コマンドにおけるAVブロックの扱いを示す図
- 【図22】「MERGE」コマンドにおけるAVブロックの扱いを示す図
- 【図23】「MERGE」コマンドにおけるAVブロックの扱いを示す図
- 【図24】DVDデコーダの構成図
- 【図25】AVデータ入力部の構成図
- 【図26】AVクリップとAVクリップ列の例を示す図
- 【図27】AVクリップとAVデータの関係を示す図

- 【図28】AVデコーダモデルの構成図
- 【図29】DVDデコーダ用のファイル構成の例を示す図
- 【図30】ビデオピクチャタイプの説明図
- 【図31】多重化の説明図
- 【図32】3つのVOBから構成されるAVファイルの例を示す図
- 【図33】拡張STDモデルの構成図
- 【図34】オーディオギャップの例を示す図
- 【図35】ビデオピクチャ構造の説明図
- 【図36】バッファ制御、オーディオギャップとオーディオ多重化境界の説明図
- 【図37】AVクリップ管理情報の構成図
- 【図38】録画後のAVクリップ管理情報の例を示す図
- 【図39】AVクリップ管理情報追加の例を示す図
- 【図40】Split前のAVクリップ管理情報の例を示す図
- 【図41】Split後のAVクリップ管理情報の例を示す図
- 【図42】Merge前のAVクリップ管理情報の例を示す図
- 【図43】Merge後のAVクリップ管理情報の例を示す図
- 【図44】シームレスフラグ判定のフローチャート
- 【図45】リモコンの例を示す図
- 【図46】初期画面の例を示す図
- 【図47】再生時のマーク設定の例を示す図
- 【図48】録画時のマーク設定の例を示す図
- 【図49】クリップ設定の例を示す図
- 【図50】クリップシーケンス設定の例を示す図
- 【図51】マーク、クリップ、クリップシーケンス削除の例を示す図
- 【図52】本編集の例を示す図
- 【図53】本編集での操作ステップの例を示す図
- 【図54】階層式編集方法のフローチャート
- 【図55】AVファイル管理情報の構成図
- 【図56】Time MapとVOBU Mapの関係を示す図
- 【図57】ストリーム情報の図
- 【図58】録画中に生成されるGOPの情報を示す図
- 【図59】録画後のAVファイル管理情報の例を示す図
- 【図60】AVファイル管理情報のSplit前の例を示す図
- 【図61】AVファイル管理情報のSplit後の例を示す図
- 【図62】AVファイル管理情報のMerge前の例を示す図
- 【図63】AVファイル管理情報のMerge後の例を示す図
- 【図64】初期及び録画時の表示の例を示す図
- 【図65】録画条件と画質選択の対応の例を示す図
- 【図66】ストリーム情報の例を示す図

63

【図67】ストリーム情報のMerge前の例を示す図

【図68】ストリーム情報のMerge後の例を示す図

【図69】AVファイル管理情報の構成図

【符号の説明】

- 100 ディスク記録部
- 101 ディスク読み出し部
- 102 ファイルシステム部
- 103 AVファイルシステム部
- 104 非AVファイルシステム部
- 105 録画・編集・再生制御部
- 106 ユーザIF部
- 107 制御データ管理部
- 110 AVデータ録画部
- 111 AVデータ入力部
- 112 AVファイル管理情報生成部
- 113 オーバーフロー対策部
- 114 AVクリップ管理情報生成部
- 120 AVデータ編集部

【図1】

## AVファイルシステム部と非AVファイルシステム部共通機能

- CREATE ファイルの作成
- DELETE ファイルの削除
- OPEN ファイルのオープン
- CLOSE ファイルのクローズ
- WRITE 非AVファイルの書き込み
- READ ファイルの読み出し(AV、非AV共通)
- SEEK データストリーム中の移動
- RENAME ファイル名の変更
- MKDIR ディレクトリの作成
- RMDIR ディレクトリの削除
- STATFS ファイルシステムの状態取得
- GET\_ATTR ファイルの属性取得
- SET\_ATTR ファイルの属性の設定
- IN\_AV\_BLK\_BOUND 指定した区間内のAVブロックバウンダリの判定
- SERACH\_DISCON\_AV\_BLK 指定された区間にAVブロック非連続境界があるか判定

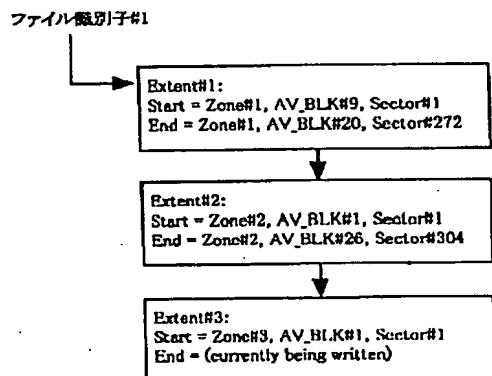
## AVファイルシステム部のみの機能

- AV-WRITE AVファイルの書き込み
- MERGE AVファイル1+バッファ+AVファイル2の結合
- SPLIT AVファイルの分割
- SHORTEN AVファイルの端部の削除
- REPLACE ファイルの部分置き換え

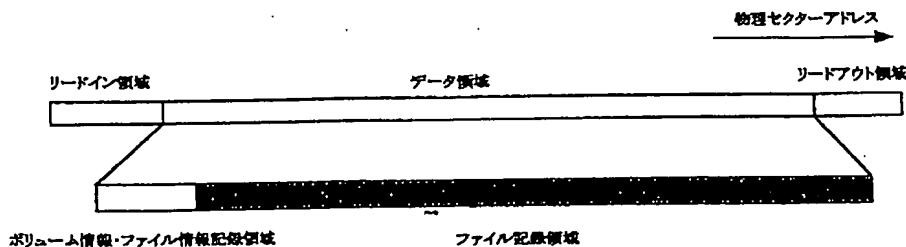
64

- 121 AVクリップ列編集部
- 122 AVクリップ編集部
- 123 AVファイル管理情報編集部
- 130 AVデータ再生部
- 131 AVデータ一部
- 132 AVクリップ列再生部
- 133 AVファイル読み出し部
- 134 読み出しエラー対策部
- 150 AVデータ入力
- 10 151 DeMUX
- 152 ビデオデコーダ
- 153 オーディオデコーダ
- 154 シームレス接続処理部
- 155 映像出力
- 156 オーディオ出力
- 200 録画時選択画面1
- 201 録画時選択画面2
- 202 録画時選択画面3

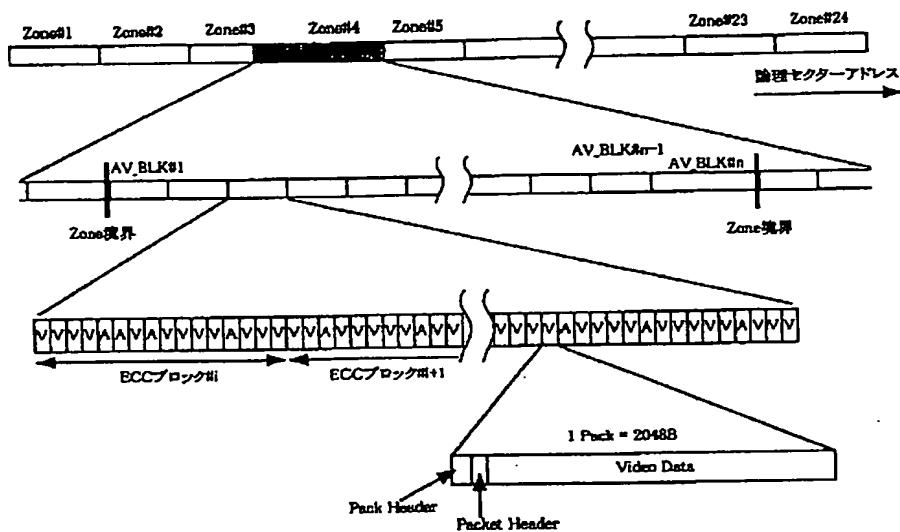
【図8】



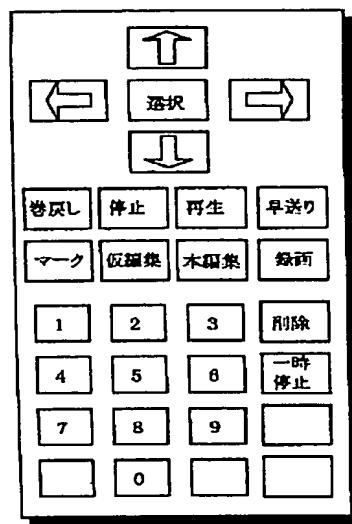
【図2】



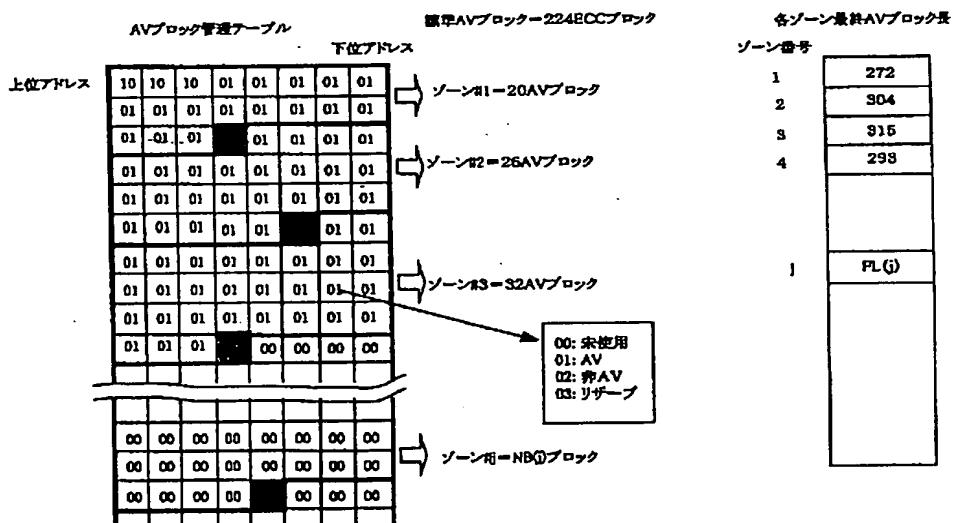
【図3】



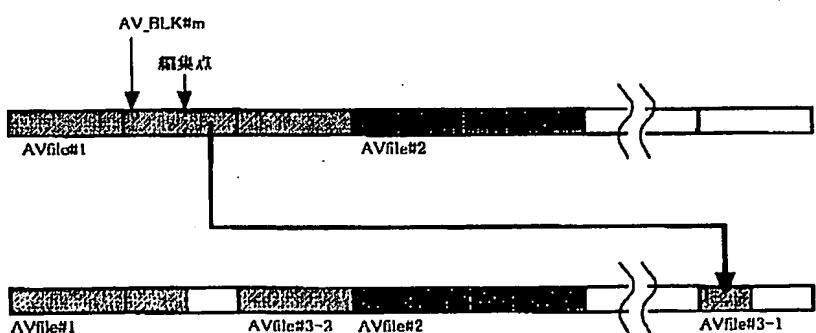
【図4.5】



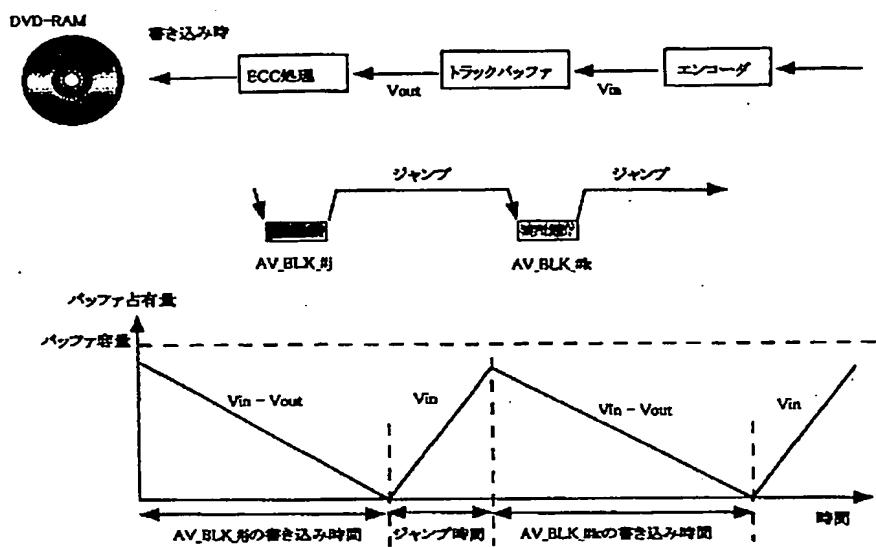
【図4】



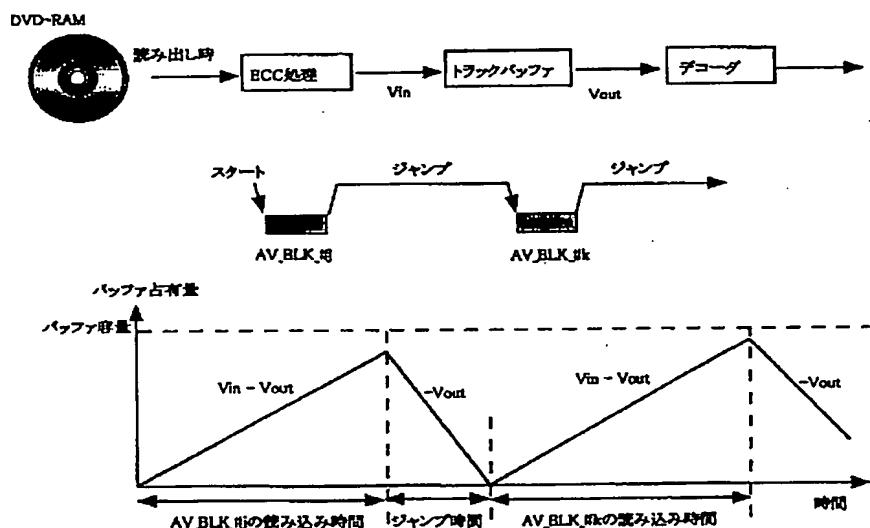
【図16】



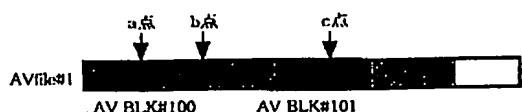
【図 5】



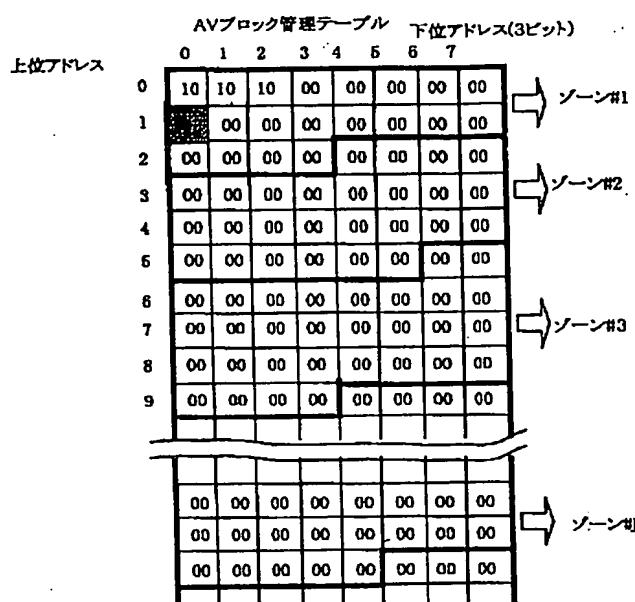
【図 6】



【図 19】

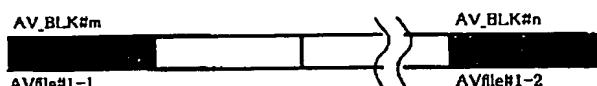


【図7】



【図20】

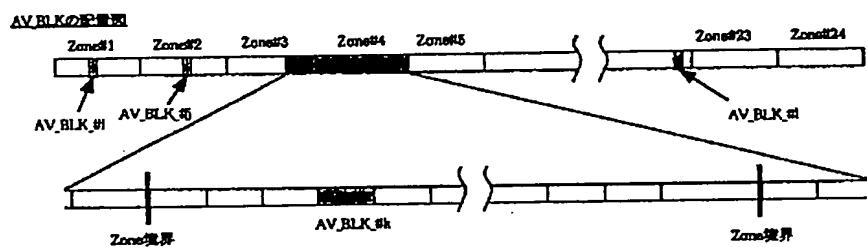
(a) AVブロックによるAVブロック不連続境界



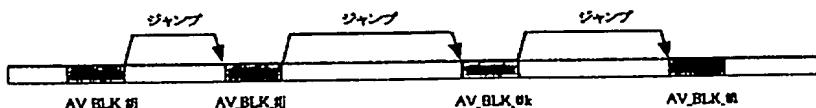
(b) ゾーン境界によるAVブロック不連続境界



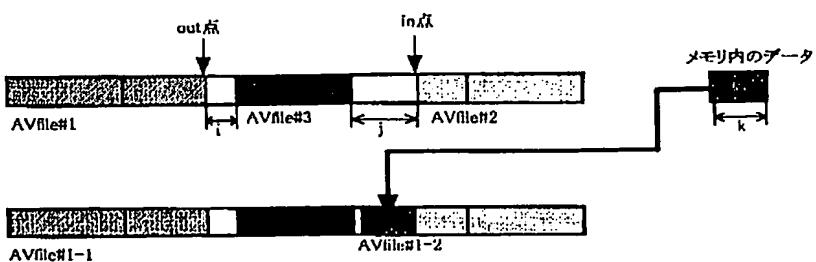
【図9】



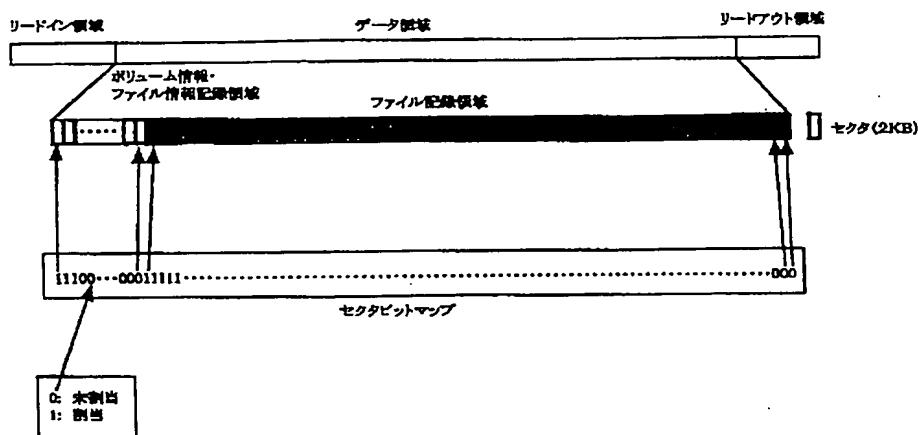
AV\_BLK#kのジャンプとサブチャンネル出力



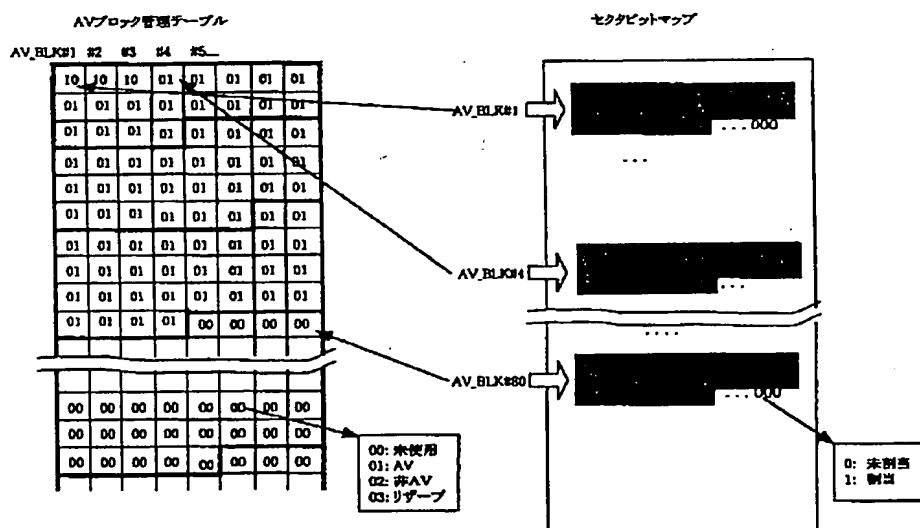
【図21】

 $i < k \leq j$ 

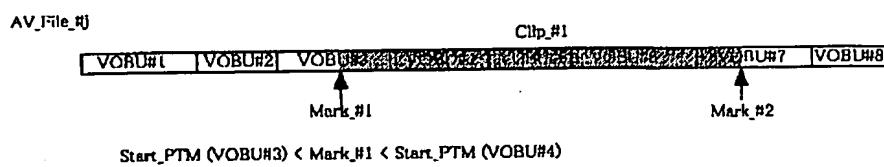
【図10】



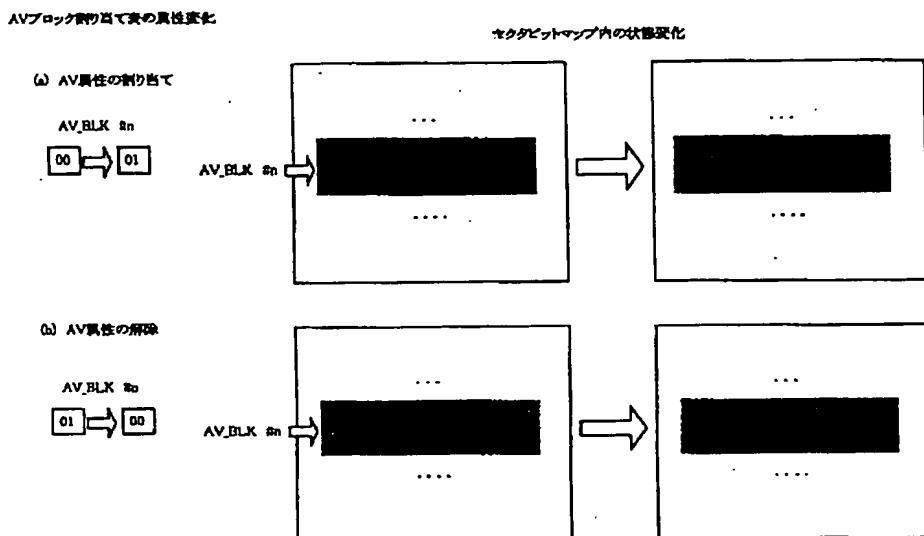
【図11】



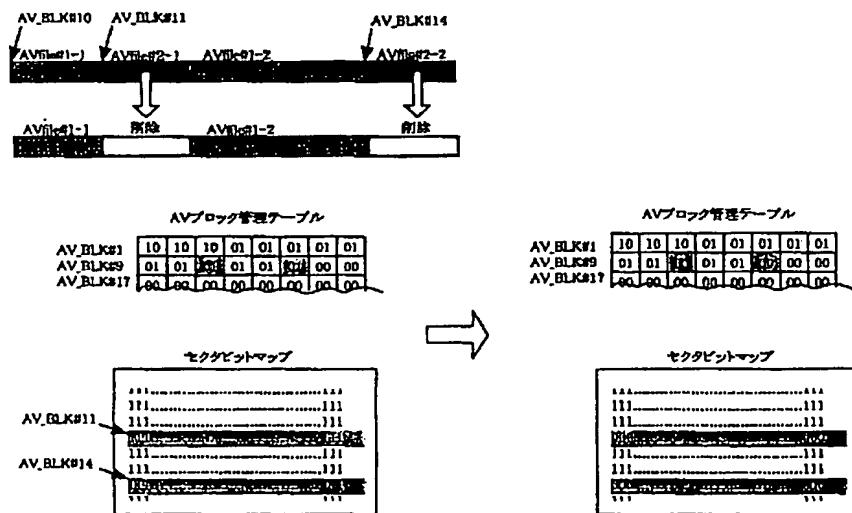
【図27】



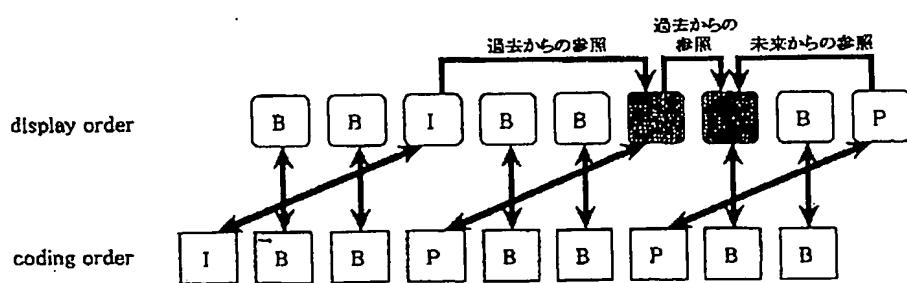
【図12】



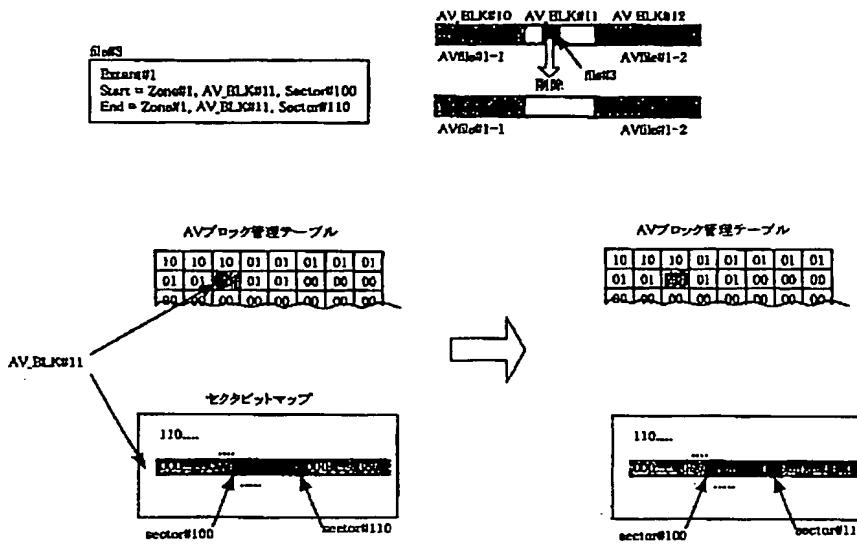
【図13】



【図30】

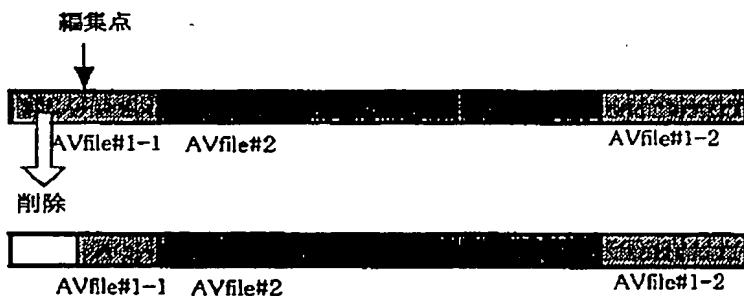


【図14】

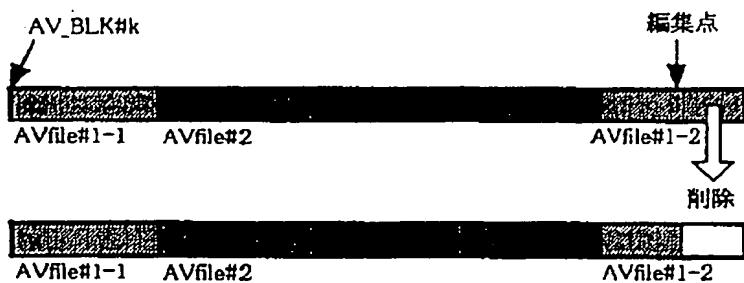


【図15】

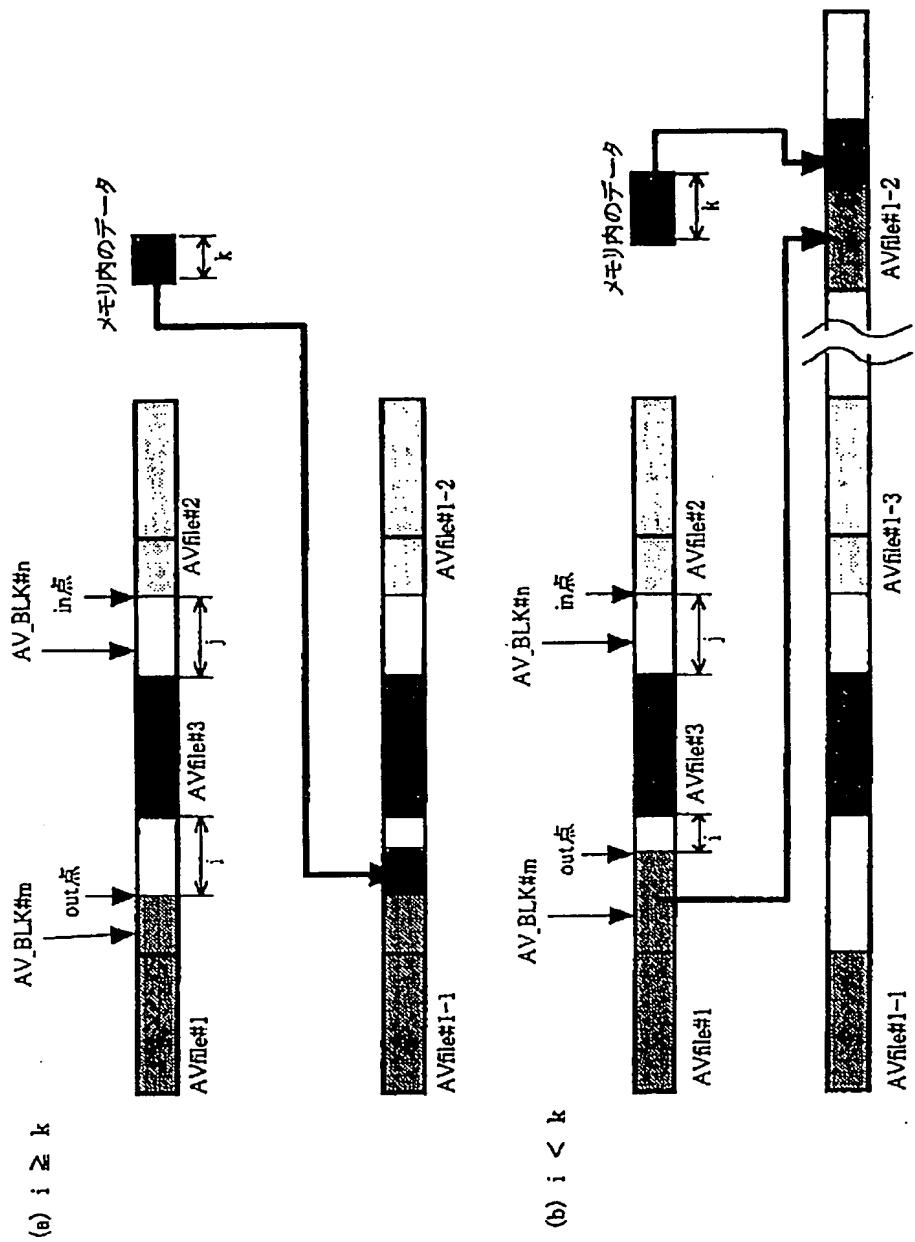
(a) AVファイルの先頭を削除する場合



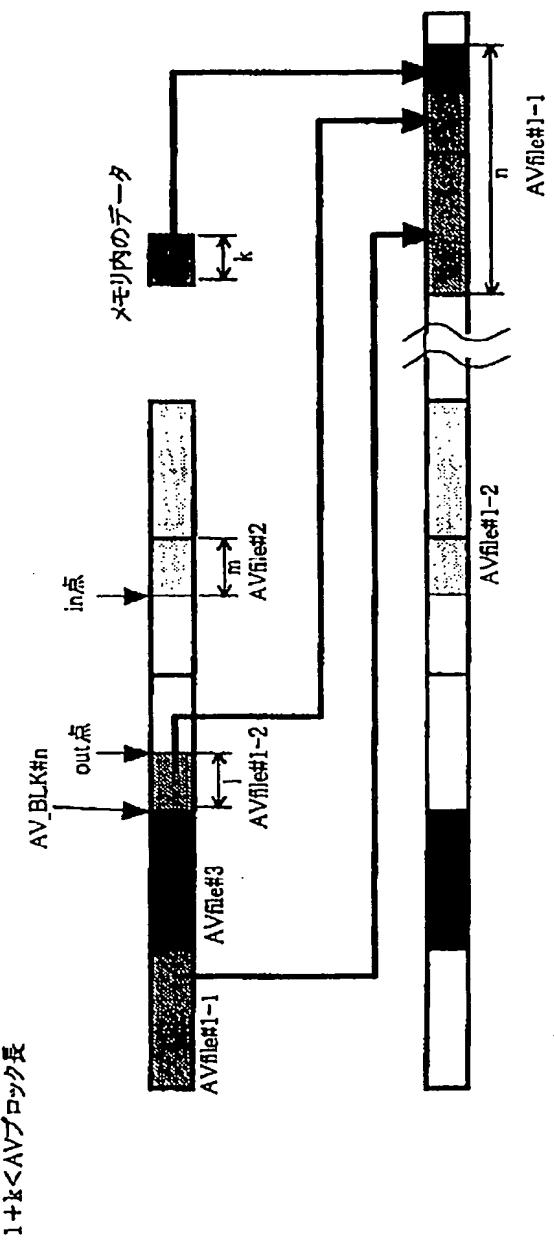
(b) AVファイルの末尾を削除する場合



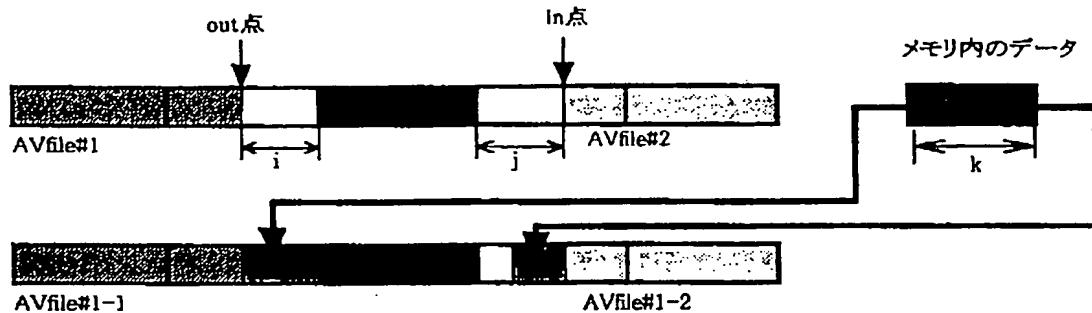
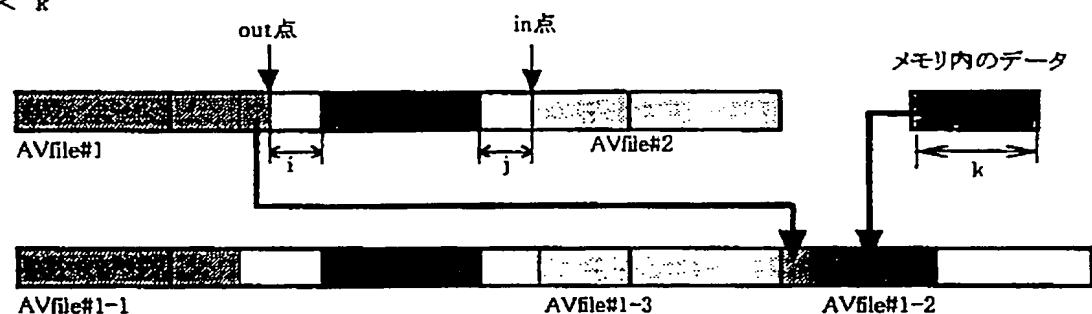
【図17】



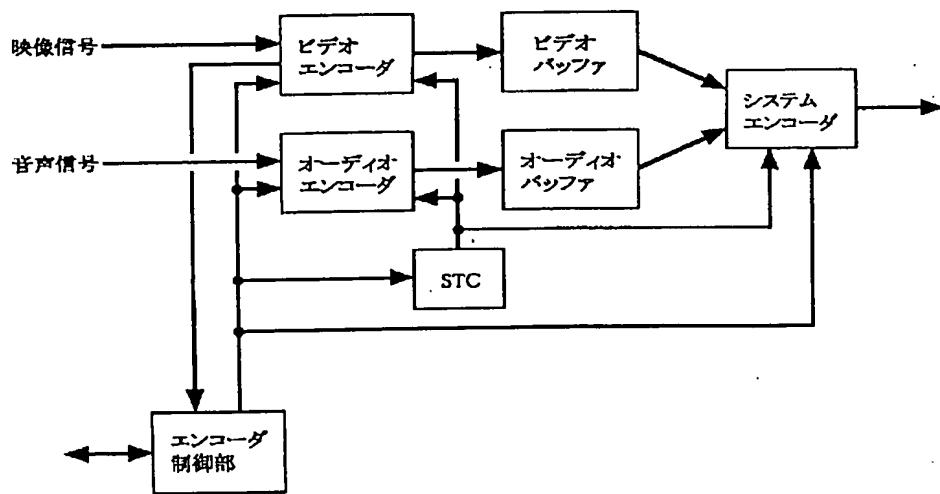
【図18】



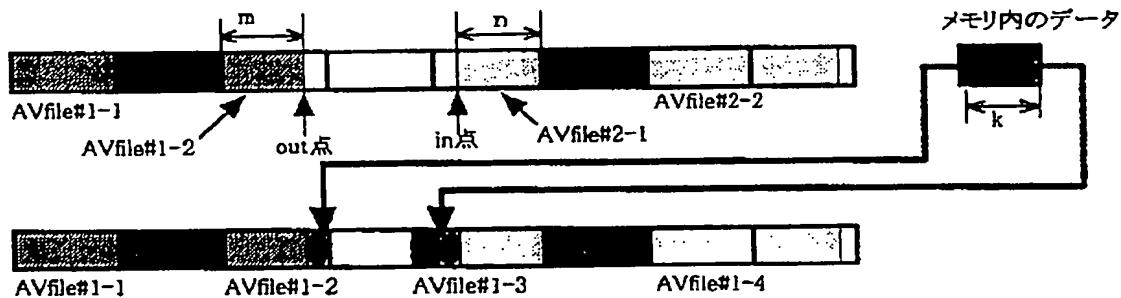
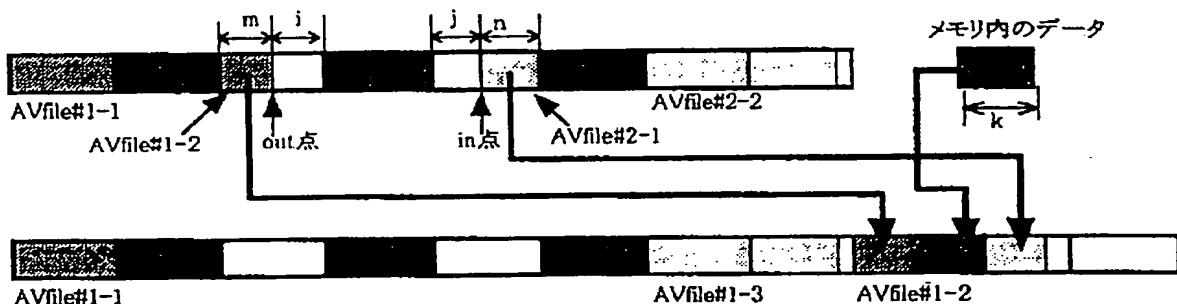
【図22】

(a)  $i + j \geq k$ ,  $i < k$ ,  $j < k$ (b)  $i + j < k$ 

【図25】

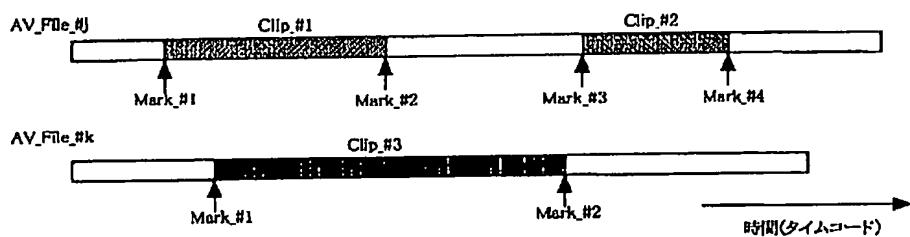


【図23】

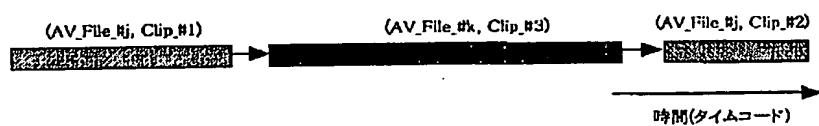
(a)  $m < \text{AVブロック長}, n < \text{AVブロック長}, 2 * \text{AVブロック長} \leq m + n + k$ (b)  $m < \text{AVブロック長}, n < \text{AVブロック長}, i + j < k, \text{AVブロック長} \leq m + n + k$ 

【図26】

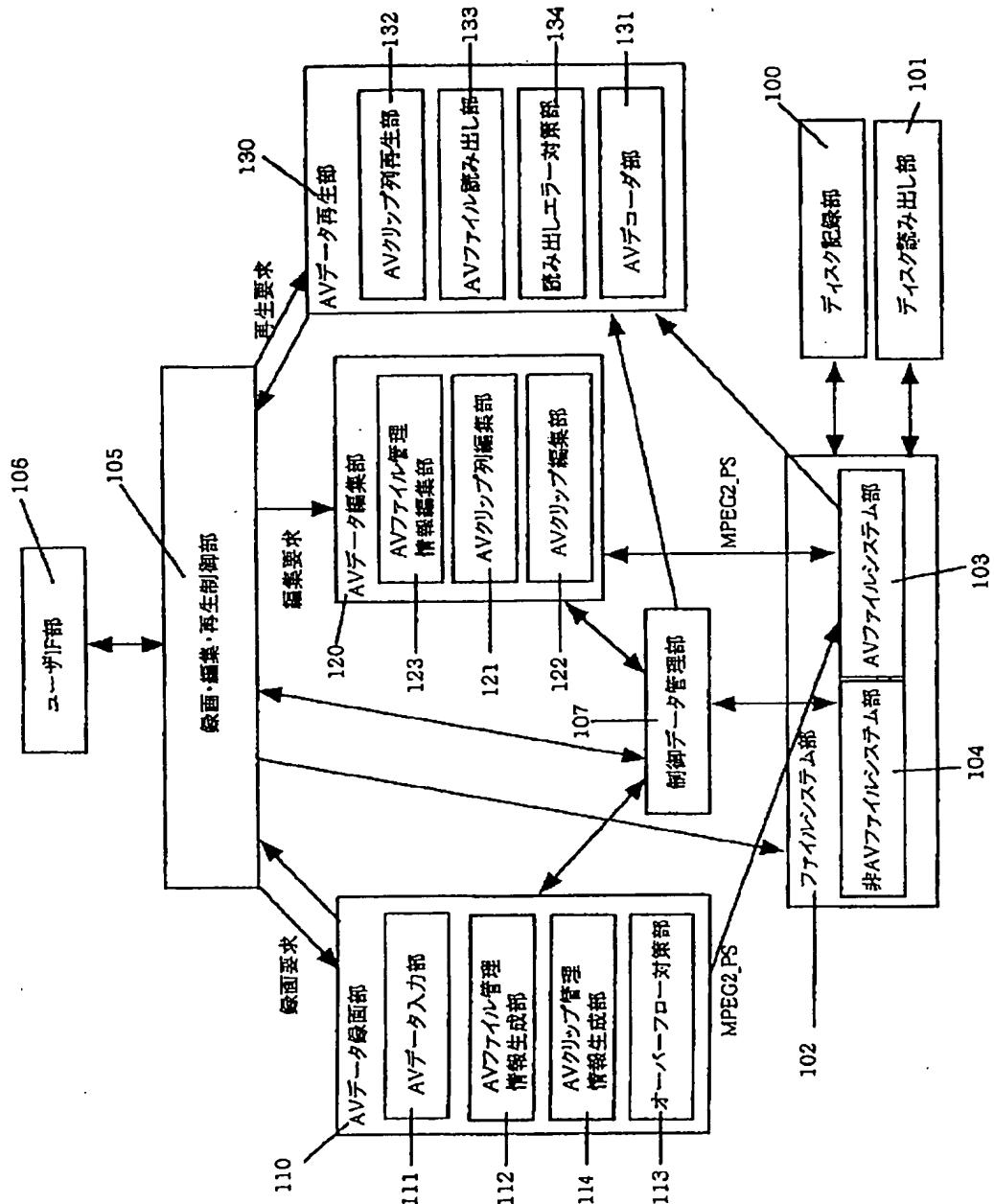
## AVクリップの例



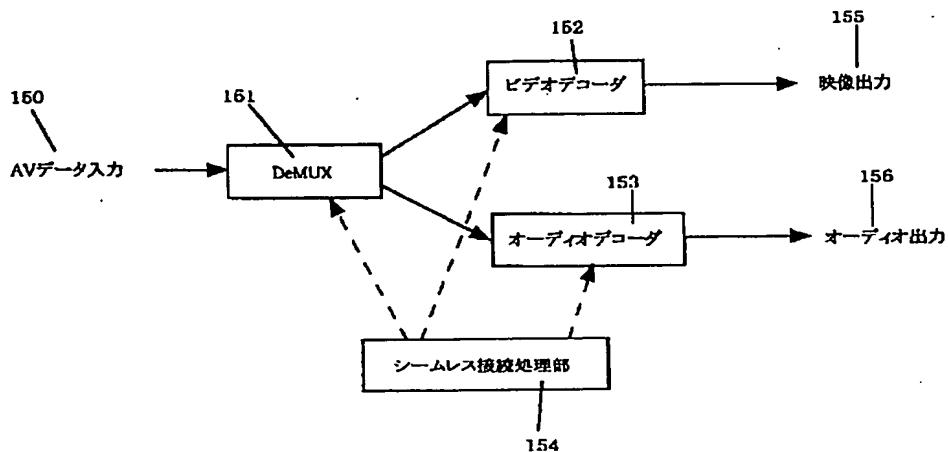
## AVクリップ列の例



【図24】



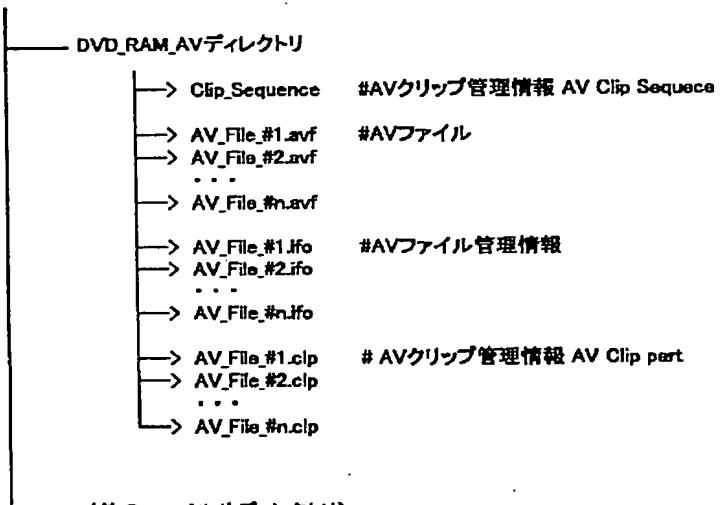
【図28】



【図29】

【図66】

ROOTディレクトリ



Stream Information

Video Attribute = (MPEG2, NTSC, 720x480, 4:3, AGC)

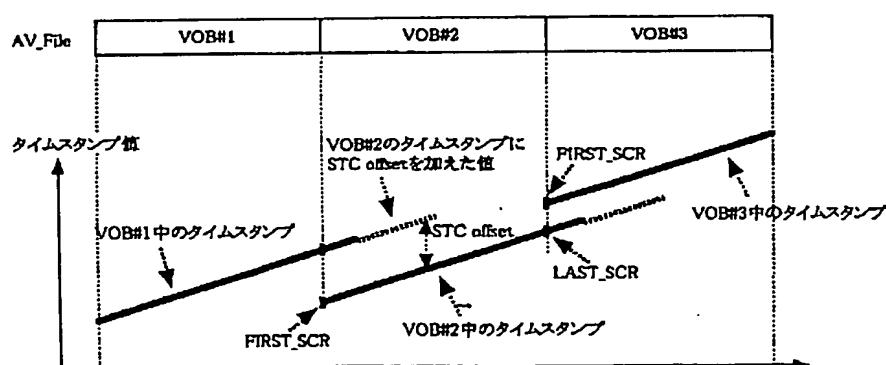
## Audio Map Table

Audio Map#1 = (Valid, 1)  
 Audio Map#2 = (Valid, 2)  
 Audio Map#3 = (Valid, 3)  
 Audio Map#4 = (Invalid)  
 :  
 Audio Map#8 = (Invalid)

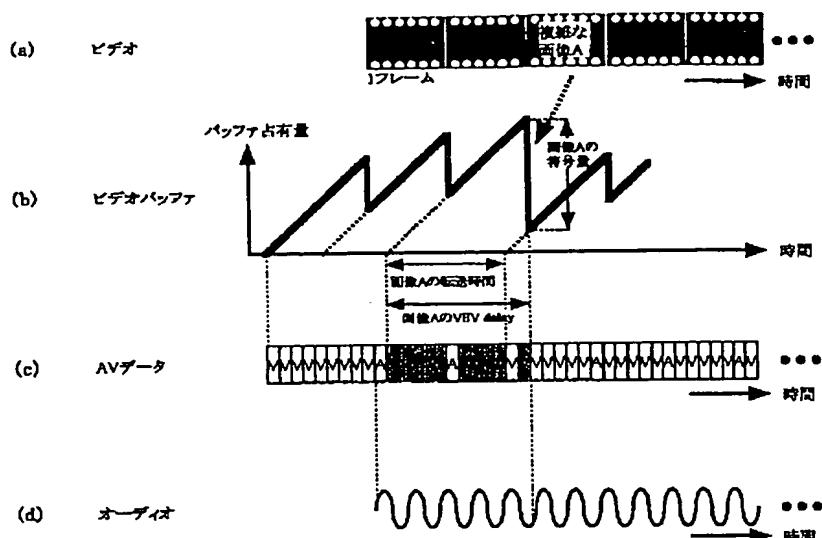
## Audio Attribute Table

Audio Attribute#1 = (Valid, MPEG1/L2, 48KHz, 192Kbps)  
 Audio Attribute#2 = (Valid, AC3, 48KHz, 384Kbps)  
 Audio Attribute#3 = (Valid, AC3, 48KHz, 384Kbps)  
 Audio Attribute#4 = (Invalid)  
 :  
 Audio Attribute#8 = (Invalid)

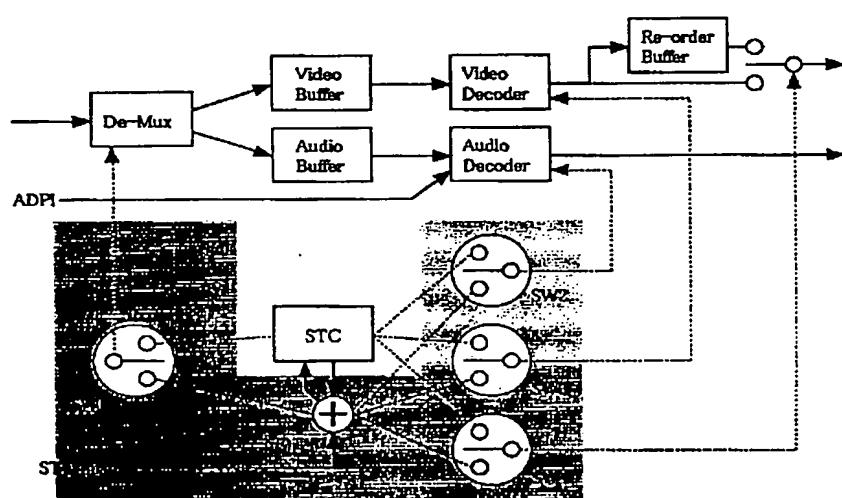
【図32】



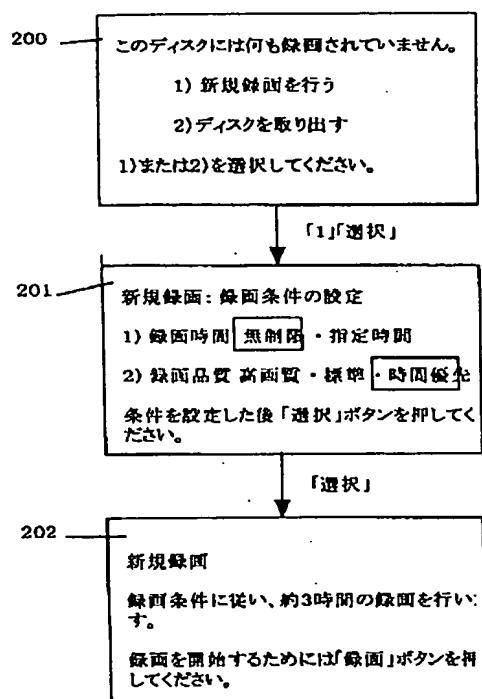
【図3-1】



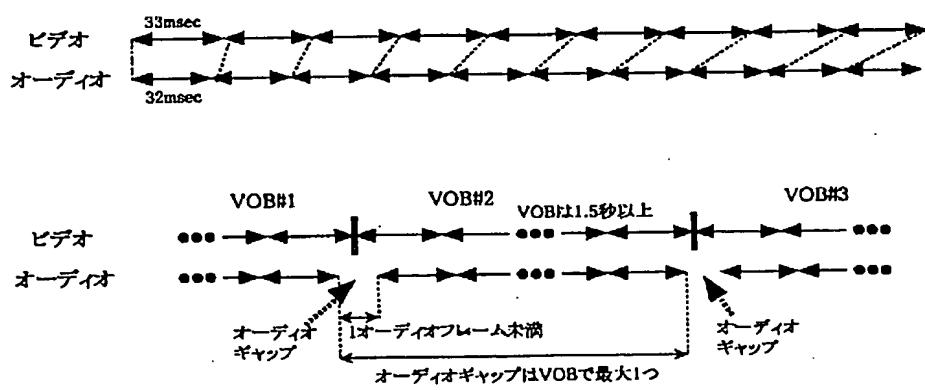
【図3-3】



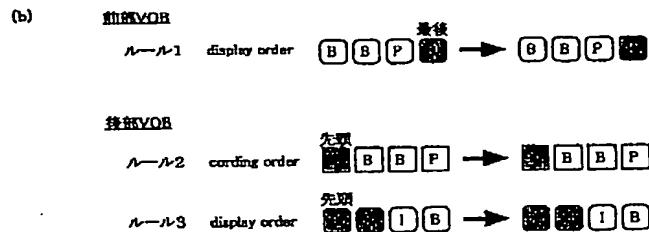
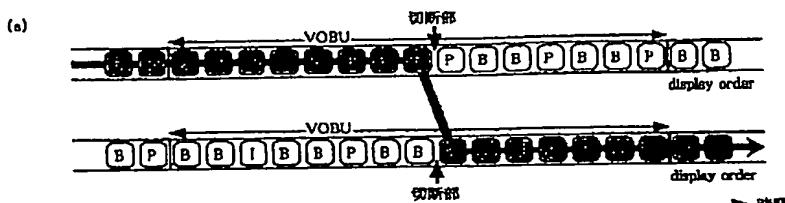
【図6-4】



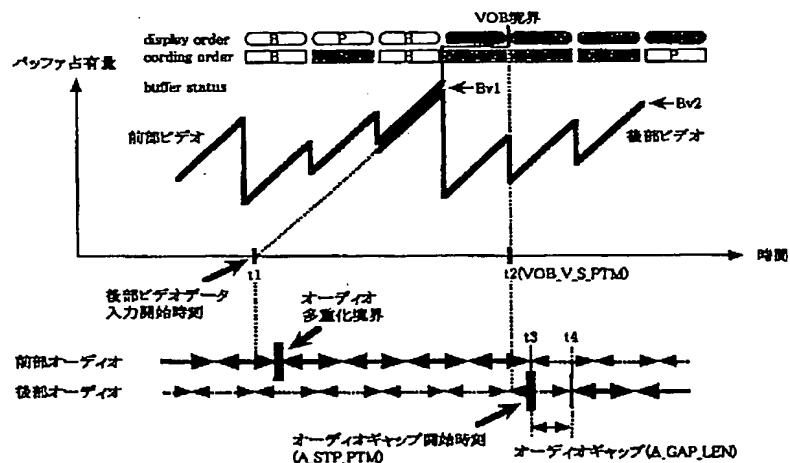
【図3-4】



【図35】



【図36】



【図58】

GOP毎の発生セクター数 I-pictureのセクター数	
GOP#1	46
GOP#2	40
GOP#3	45
GOP#4	53
GOP#5	49
GOP#6	44
GOP#7	45
GOP#10700	46
	14
	15
	12
	17
	15
	13
	13
	15
	13
	17
	17
	12
	14

(但し、GOPの構成は16フレームに固定)

【図38】

(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=1)

```

Clip_Information = (VOB_ID="1", Start_Time="00:00:00:00", VOB_V_S_PTSM=90090, VOB_V_E_PTSM=405405,
FIRST_SCR=0, LAST_SCR=305405)

Seamless_Information=(Seamless_Flag="NONE", A_STP_PTSM=0, A_GAP_LEN=0)

Mark_Table = (Num_of_Marks=2)

Mark#1 = (00:00:00:00)
Mark#2 = (00:00:08:10)

Clip_Table = (Num_of_Clips=1)

Clip#1 = (Start_Mark=Mark#1, End_Mark=Mark#2)

```

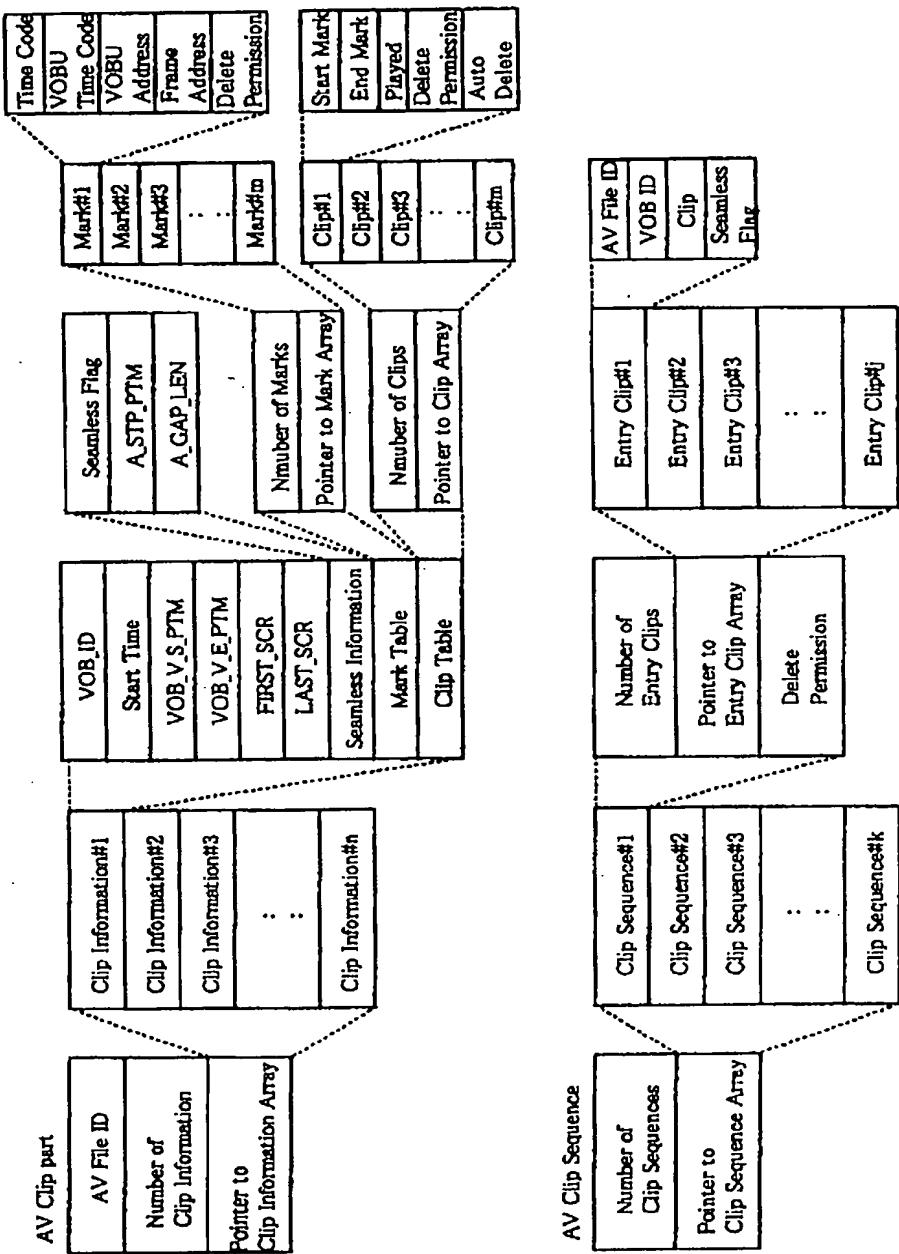
(b) AV Clip Sequence = (Num\_of\_Clip\_Sequences=1)

```

Clip_Sequence#1 = (Num_of_Entry_Clips=1) ...
Entry_Clip#1 = (AV_File_ID="System_2", VOB_ID="1", Clip=Clip#1, Seamless_Flag="NONE")

```

【図 37】



【図 3 9】

(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=1)

Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTSM=90090, VOB\_V\_E\_PTSM=405405  
FIRST\_SCR=0, LAST\_SCR=305405)

Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTSM=0, A\_GAP\_LEN=0)

Mark\_Table = (Num\_of\_Marks=2)

Mark#1 = (00:00:00:00)  
Mark#2 = (00:00:03:14)  
Mark#3 = (00:00:01:00)

Clip\_Table = (Num\_of\_Clips=1)

Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)  
Clip#2 = (Start\_Mark=Mark#3, End\_Mark=Mark#2)

(b) AV Clip Sequence = (Num\_of\_Clip\_Sequences=2)

Clip\_Sequence#1 = (Num\_of\_Entry\_Clips=1)

Entry\_Clip#1 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")

Clip\_Sequence#2 = (Num\_of\_Entry\_Clips=1)

Entry\_Clip#2 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#2, Seamless\_Flag="NONE")

【図 4 0】

(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=1)

Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTSM=90090, VOB\_V\_E\_PTSM=405405  
FIRST\_SCR=0, LAST\_SCR=305405)

Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTSM=0, A\_GAP\_LEN=0)

Mark\_Table = (Num\_of\_Marks=2)

Mark#1 = (00:00:00:00)  
Mark#2 = (00:00:03:14)

Clip\_Table = (Num\_of\_Clips=1)

Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

(b) AV Clip Sequence = (Num\_of\_Clip\_Sequences=1)

Clip\_Sequence#1 = (Num\_of\_Entry\_Clips=1)

Entry\_Clip#1 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")

【図 4 1】

(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=1)

Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTSM=90090, VOB\_V\_E\_PTSM=270270  
FIRST\_SCR=0, LAST\_SCR=180270)

Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTSM=0, A\_GAP\_LEN=0)

Mark\_Table = (Num\_of\_Marks=2)

Mark#1 = (00:00:00:00)  
Mark#2 = (00:00:01:29)

Clip\_Table = (Num\_of\_Clips=1)

Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

(b) AV Clip part = (AV\_File\_ID = "System\_3", Num\_of\_Clip\_Information=1)

Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTSM=270270, VOB\_V\_E\_PTSM=405405  
FIRST\_SCR=170270 LAST\_SCR=305405)

Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTSM=0, A\_GAP\_LEN=0)

Mark\_Table = (Num\_of\_Marks=2)

Mark#1 = (00:00:00:00)  
Mark#2 = (00:00:01:14)

Clip\_Table = (Num\_of\_Clips=1)

Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

(c) AV Clip Sequence = (Num\_of\_Clip\_Sequences=2)

Clip\_Sequence#1 = (Num\_of\_Entry\_Clips=1)

Entry\_Clip#1 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")

Clip\_Sequence#2 = (Num\_of\_Entry\_Clips=1)

Entry\_Clip#2 = (AV\_File\_ID="System\_3", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")

【図4-2】

(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=1)  
 Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTm=90090, VOB\_V\_E\_PTm=270270  
 FIRST\_SCR=0, LAST\_SCR=170270)  
 Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTm=0, A\_GAP\_LEN=0)  
 Mark\_Table = (Num\_of\_Marks=2)  
 Mark#1 = (00:00:00:00)  
 Mark#2 = (00:00:01:29)  
 Clip\_Table = (Num\_of\_Clips=1)  
 Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

(b) AV Clip part = (AV\_File\_ID = "System\_3", Num\_of\_Clip\_Information=1)  
 Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTm=270270, VOB\_V\_E\_PTm=405405  
 FIRST\_SCR=170270, LAST\_SCR=305405)  
 Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTm=0, A\_GAP\_LEN=0)  
 Mark\_Table = (Num\_of\_Marks=2)  
 Mark#1 = (00:00:00:00)  
 Mark#2 = (00:00:01:14)  
 Clip\_Table = (Num\_of\_Clips=1)  
 Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

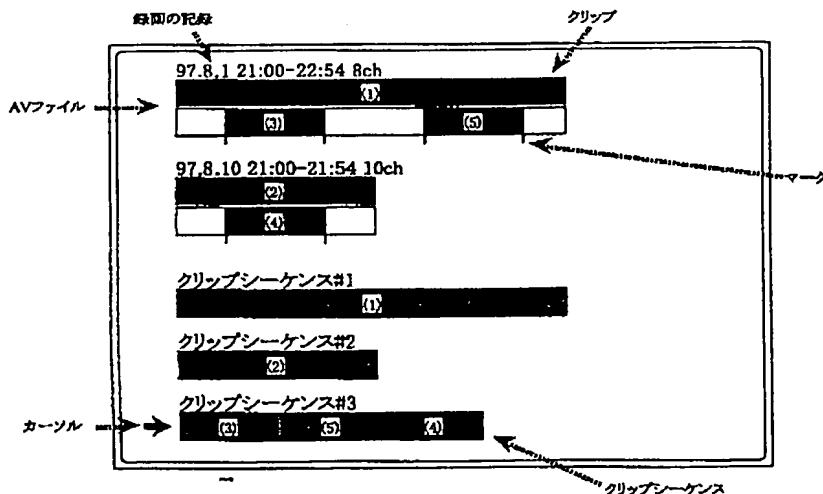
(c) AV Clip Sequence = (Num\_of\_Clip\_Sequences=2)  
 Clip\_Sequence#1 = (Num\_of\_Entry\_Clips=2)  
 Entry\_Clip#1 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")  
 Entry\_Clip#2 = (AV\_File\_ID="System\_3", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")

【図4-3】

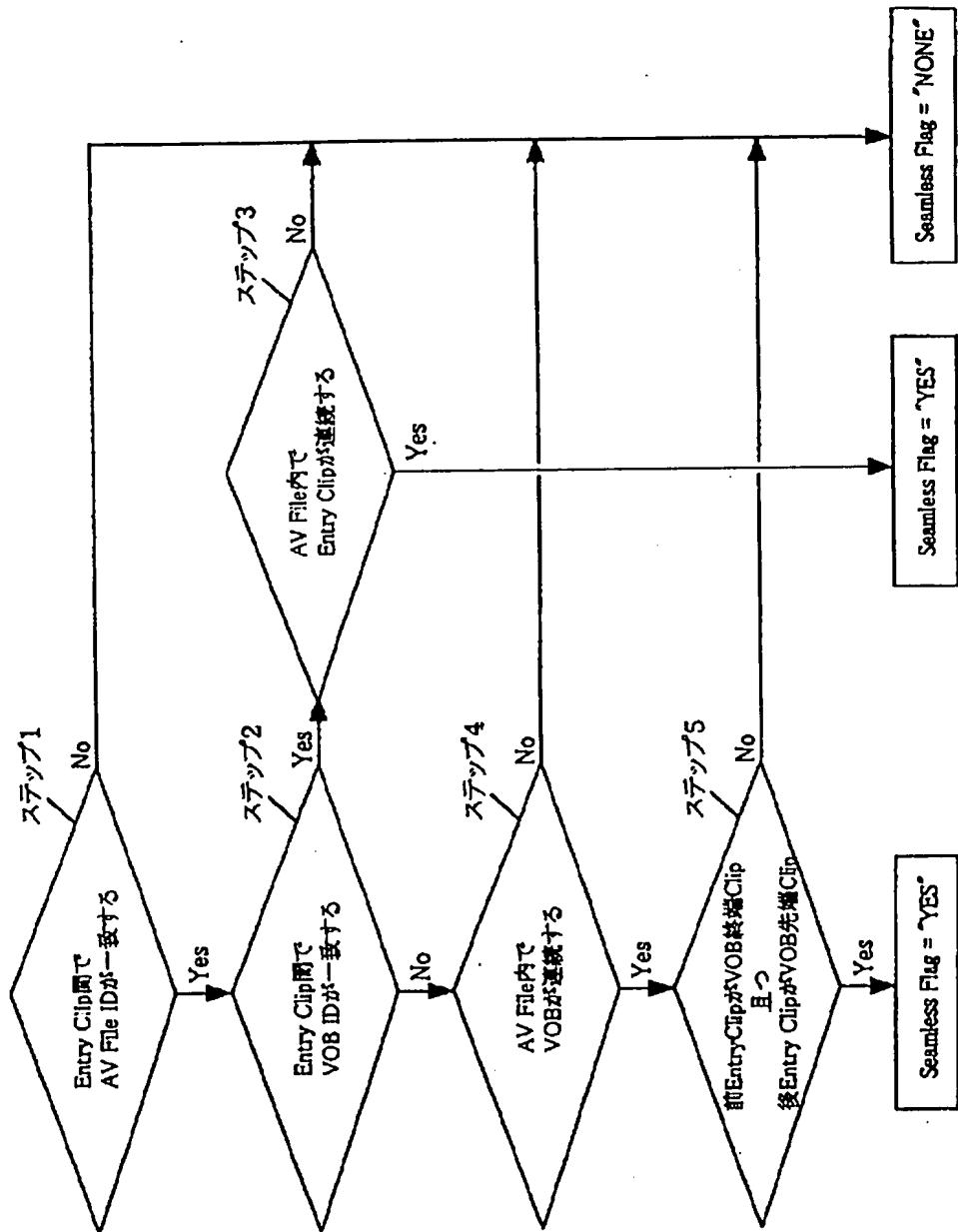
(a) AV Clip part = (AV\_File\_ID = "System\_2", Num\_of\_Clip\_Information=2)  
 Clip\_Information = (VOB\_ID="1", Start\_Time="00:00:00:00", VOB\_V\_S\_PTm=90090, VOB\_V\_E\_PTm=306306  
 FIRST\_SCR=0, LAST\_SCR=206306)  
 Seamless\_Information=(Seamless\_Flag="NONE", A\_STP\_PTm=0, A\_GAP\_LEN=0)  
 Mark\_Table = (Num\_of\_Marks=2)  
 Mark#1 = (00:00:00:00)  
 Mark#2 = (00:00:02:11)  
 Clip\_Table = (Num\_of\_Clips=1)  
 Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)  
 Clip\_Information = (VOB\_ID="2", Start\_Time="00:00:02:12", VOB\_V\_S\_PTm=216216, VOB\_V\_E\_PTm=405405  
 FIRST\_SCR=116216, LAST\_SCR=305405)  
 Seamless\_Information=(Seamless\_Flag="YES", A\_STP\_PTm=216810, A\_GAP\_LEN=378)  
 Mark\_Table = (Num\_of\_Marks=2)  
 Mark#1 = (00:00:00:00)  
 Mark#2 = (00:00:02:12)  
 Clip\_Table = (Num\_of\_Clips=1)  
 Clip#1 = (Start\_Mark=Mark#1, End\_Mark=Mark#2)

(b) AV Clip Sequence = (Num\_of\_Clip\_Sequences=1)  
 Clip\_Sequence#1 = (Num\_of\_Entry\_Clips=2)  
 Entry\_Clip#1 = (AV\_File\_ID="System\_2", VOB\_ID="1", Clip=Clip#1, Seamless\_Flag="NONE")  
 Entry\_Clip#2 = (AV\_File\_ID="System\_2", VOB\_ID="2", Clip=Clip#1, Seamless\_Flag="YES")

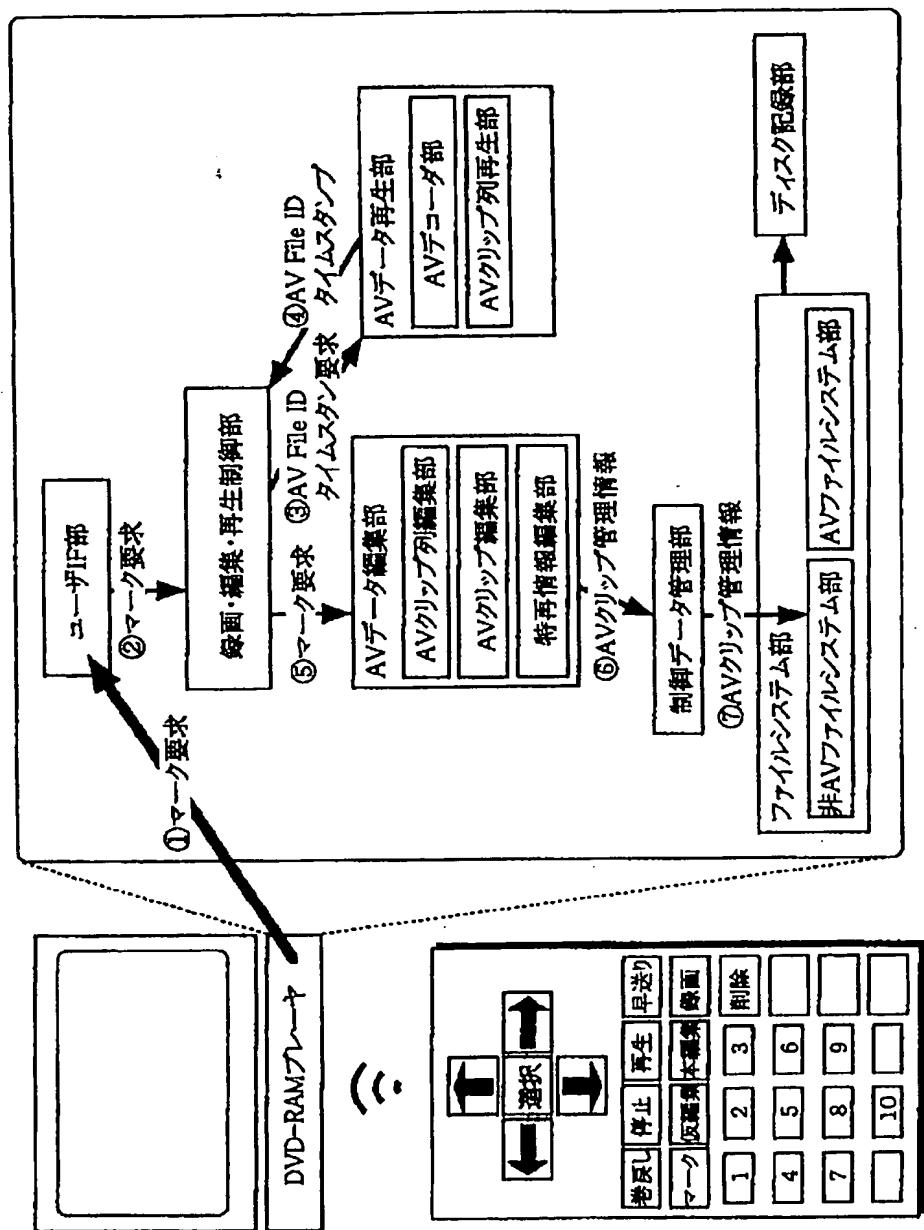
【図4-6】



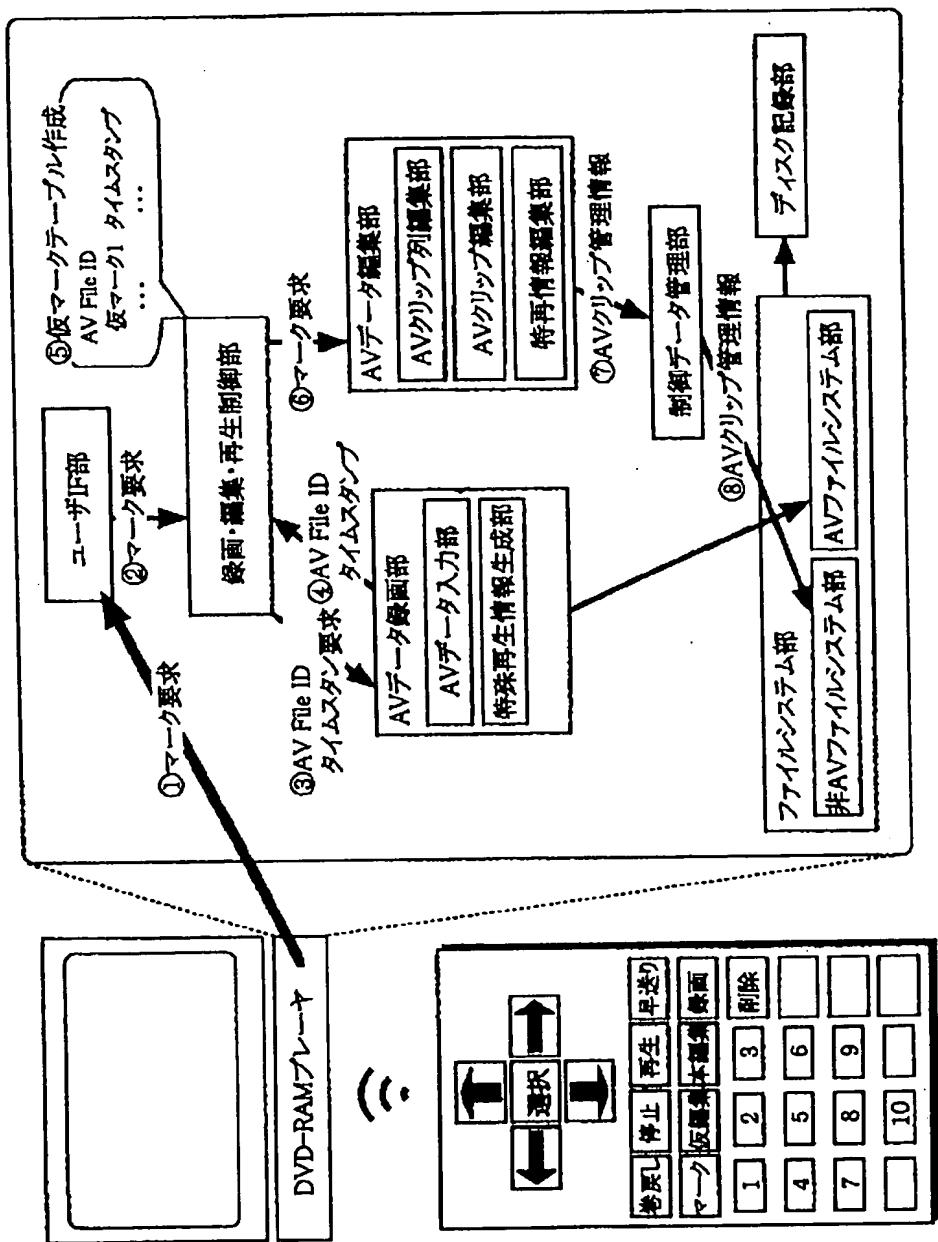
【図4-4】



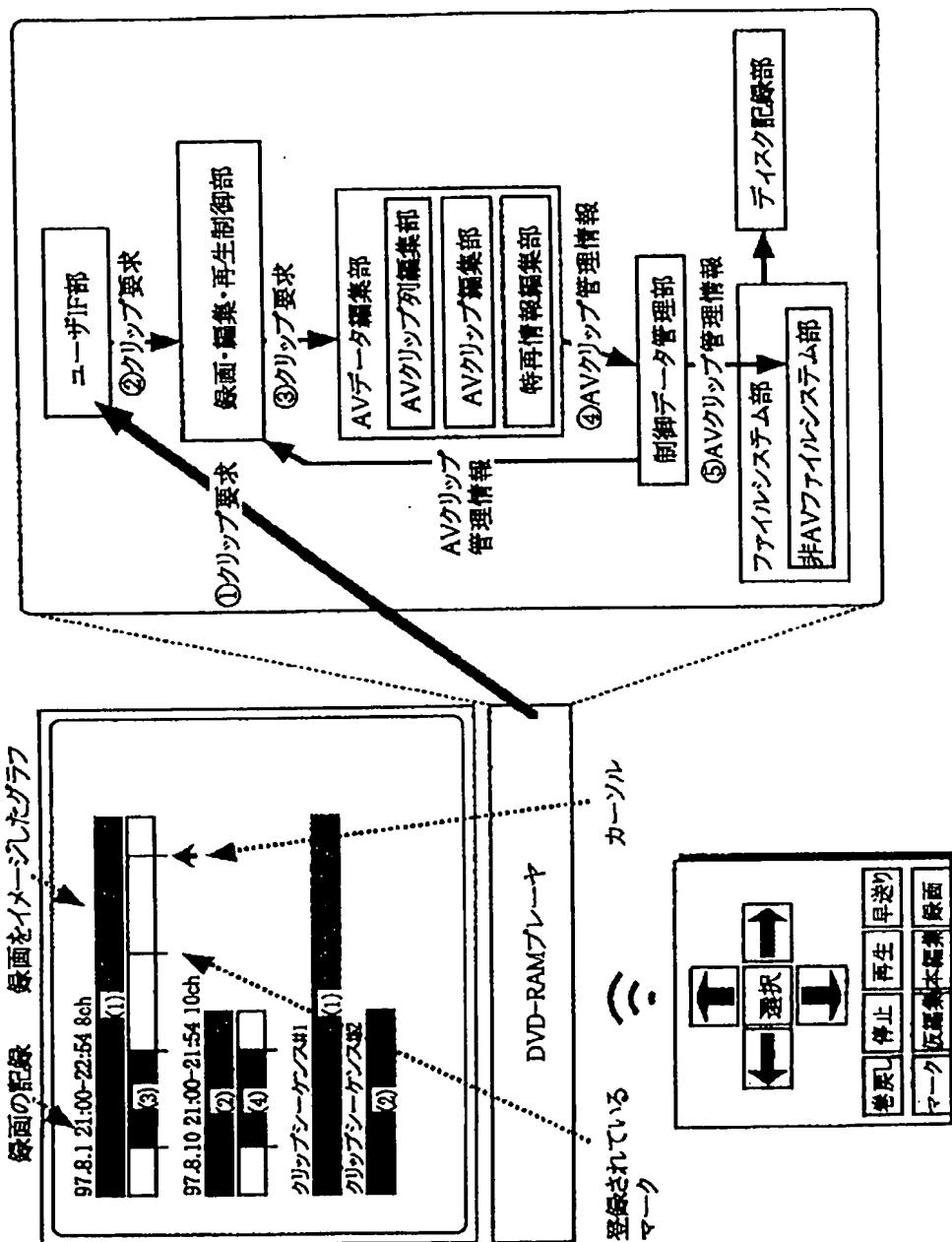
【図47】



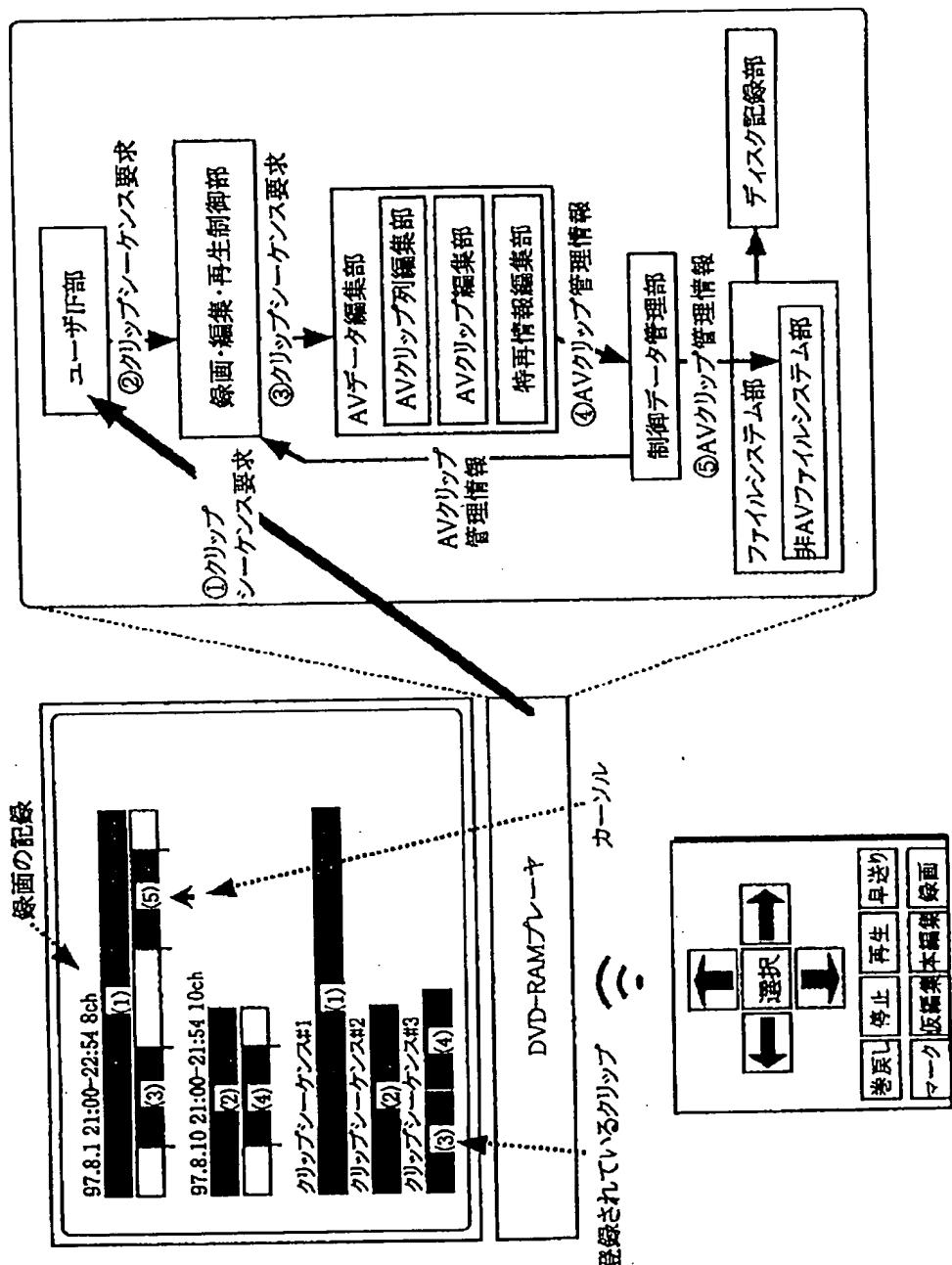
【図48】



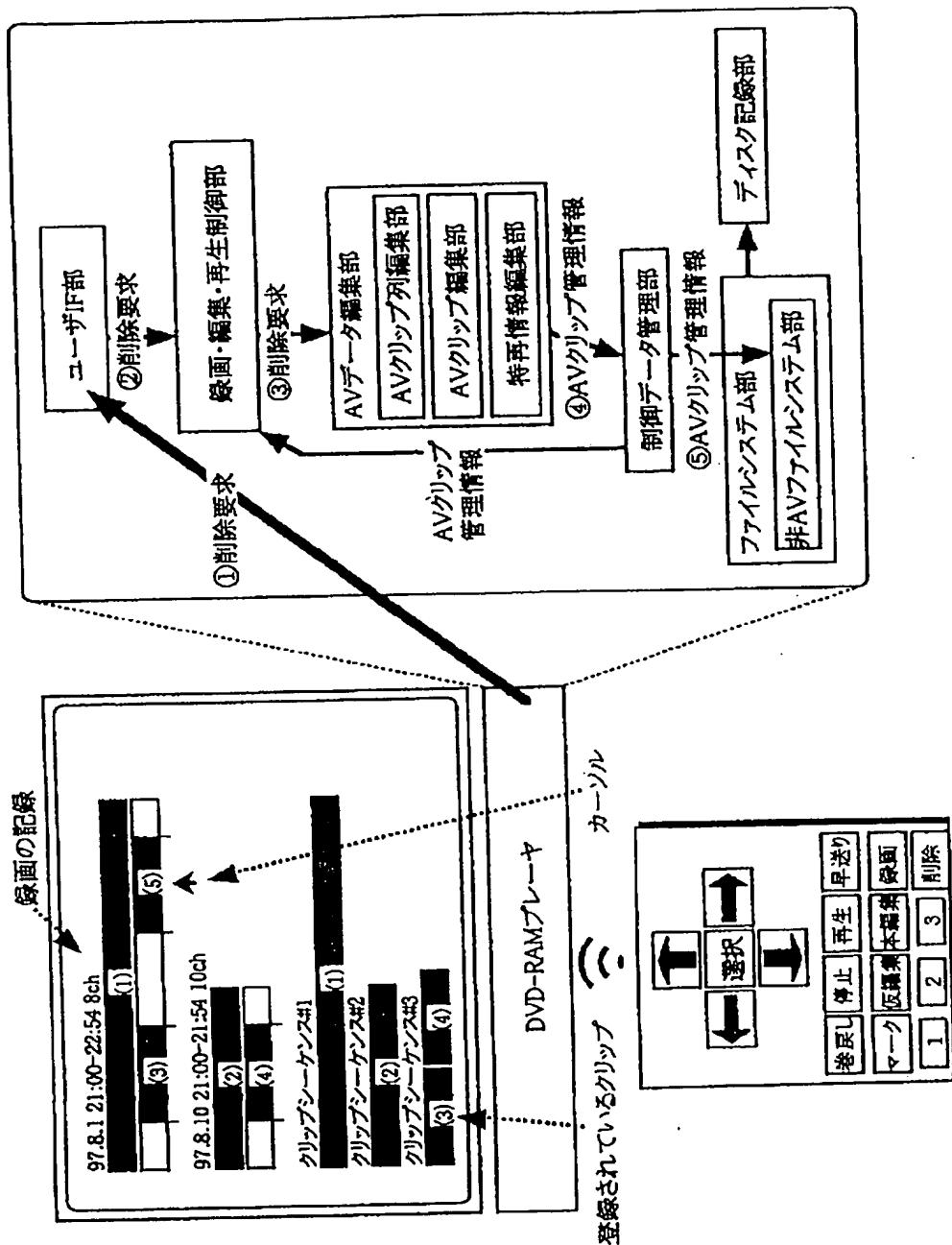
【図49】



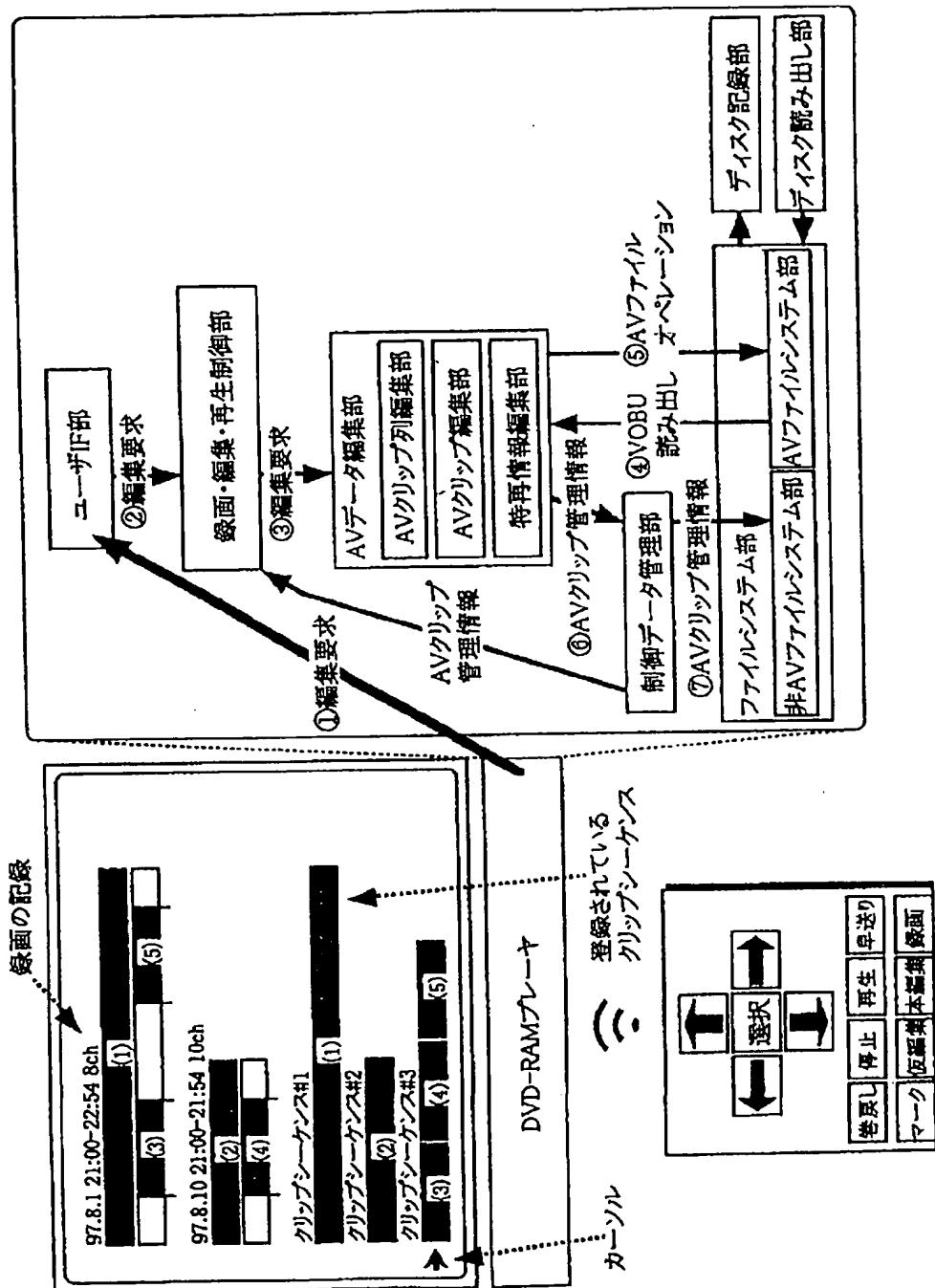
【図50】



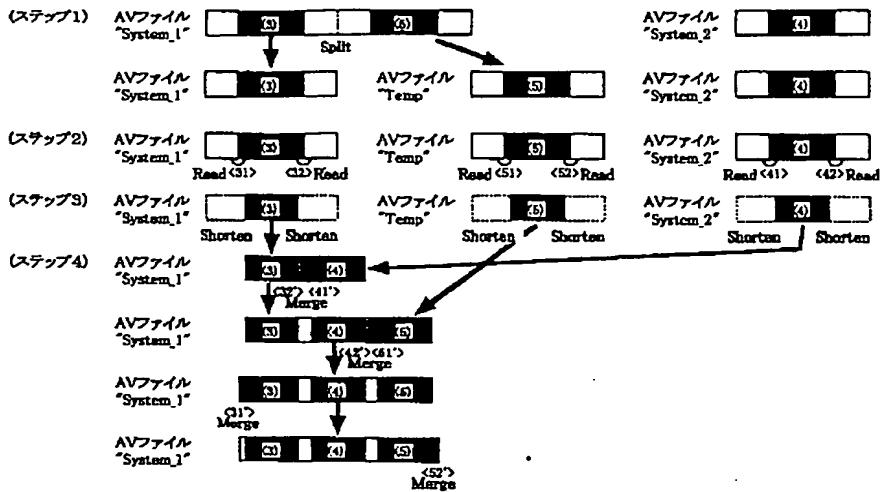
【図51】



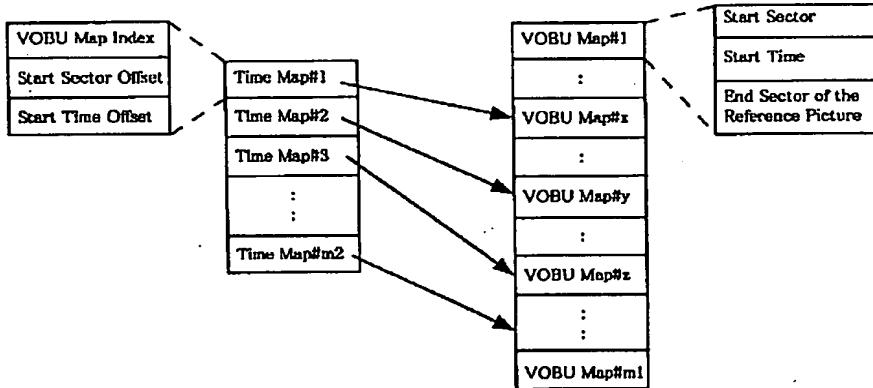
【図52】



【図 5 3】



【図 5 6】



【図 5 9】

```

AV File Manager Information = (AV_File_ID = "System_1", Sector Size = 49306, Playback Time = 01:27:50:00)
VOB_Table = (Num_of_VOB = 1)

VOB_Information#1 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = 0, Time_Offset = 00:00:00:00)

VOBU_Map_Table = (Num_of_Map = 10700)

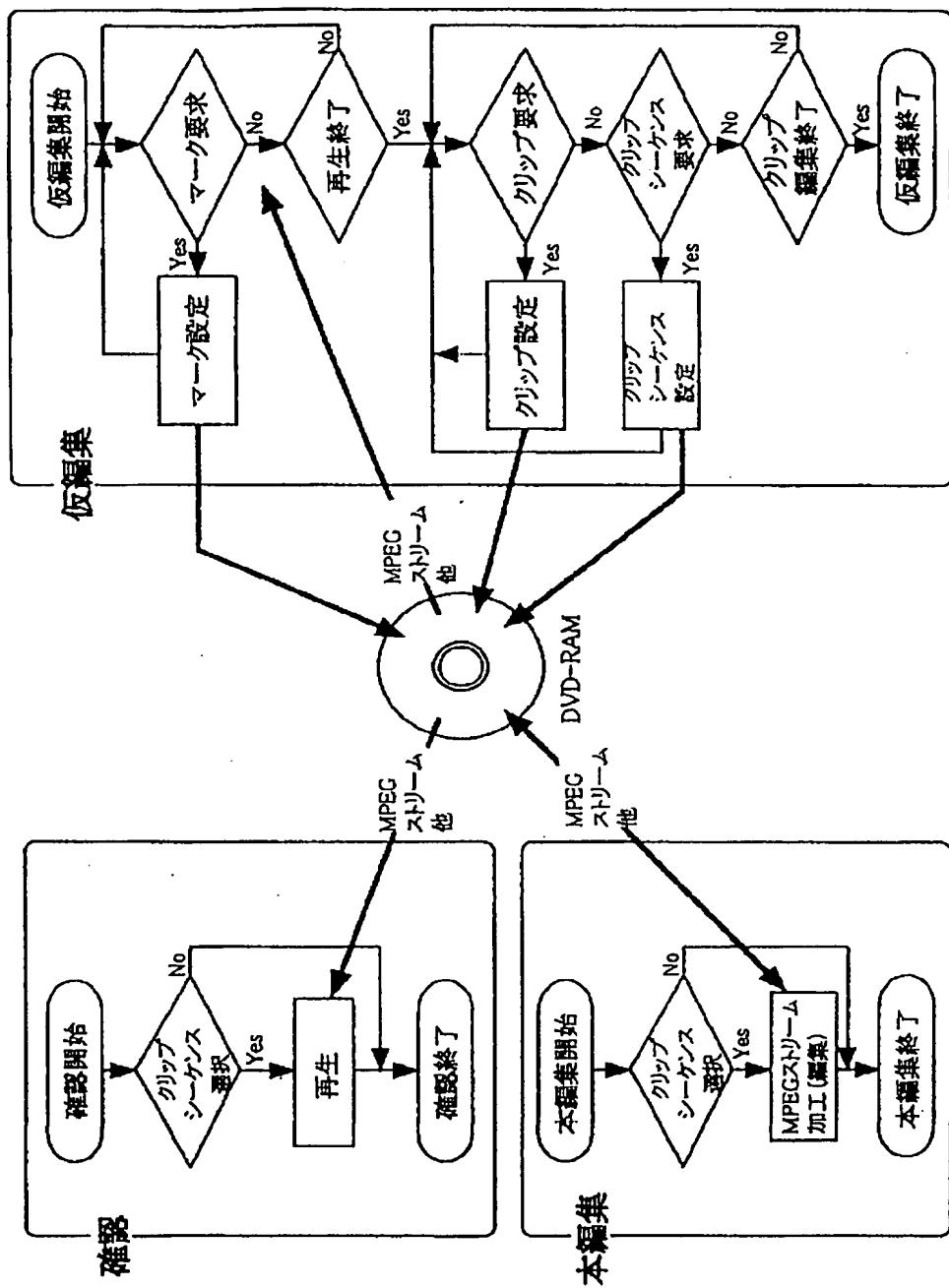
VOBU_Map#1 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 13)
VOBU_Map#2 = (Start_Sector = 46, Start_Time = 00:15, Ref_Pict_EA = 14)
VOBU_Map#3 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 11)
VOBU_Map#4 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 16)
VOBU_Map#5 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 14)
VOBU_Map#6 = (Start_Sector = 49, Start_Time = 00:15, Ref_Pict_EA = 12)
VOBU_Map#7 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 12)
...
VOBU_Map#10700 = (Start_Sector = 46, Start_Time = 00:15, Ref_Pict_EA = 13)

Time_Map_Table = (Num_of_Map = 5377, Time_Unit = 1, Time_Base = 00:00, Index_Offset = 0)

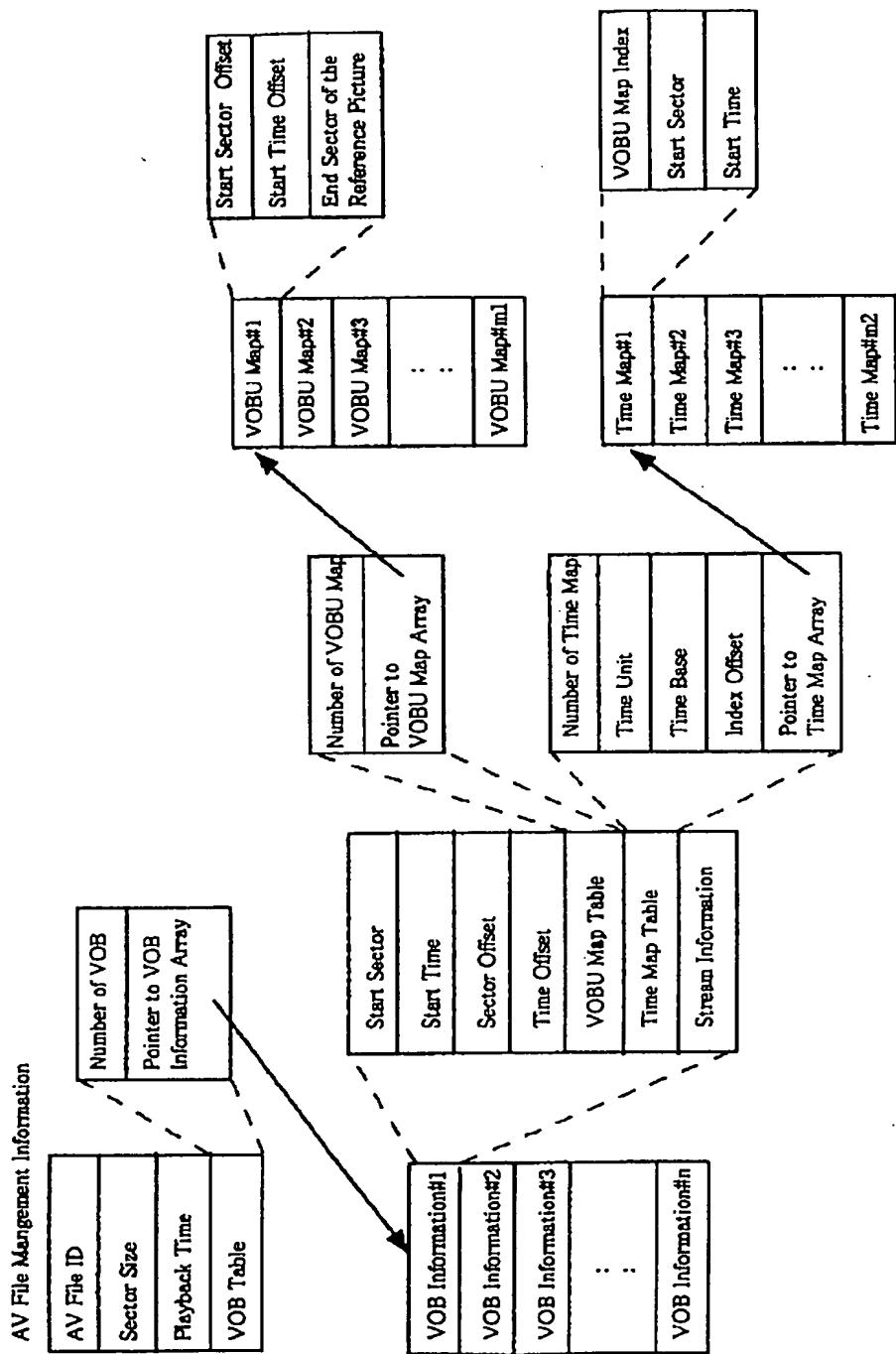
Time_Map#1 = (VOBU_Map_Index = 3, Start_Sector = 86, Start_Time = 00:00:01:00)
Time_Map#2 = (VOBU_Map_Index = 5, Start_Sector = 184, Start_Time = 00:00:02:00)
Time_Map#3 = (VOBU_Map_Index = 7, Start_Sector = 277, Start_Time = 00:00:03:00)
...
Time_Map#5377 = (VOBU_Map_Index = 10699)

```

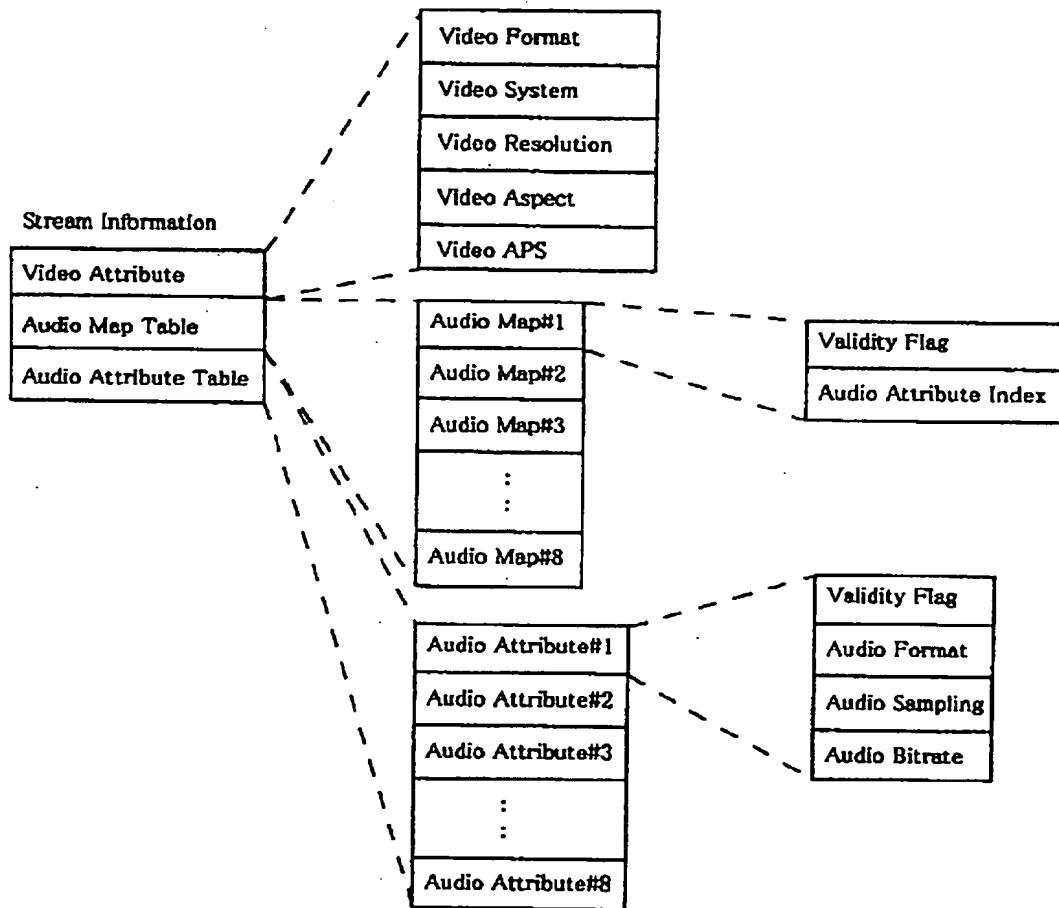
【図54】



[図55]



【図 57】



【図 60】

AV File Manager Information = (AV\_File\_ID = "System\_2", Sector\_Size= 322, Playback\_Time = 00:00:03:15)

VOB Table = (Num\_of\_VOB = 1)

VOB\_Information#1 = (Start\_Sector = 0, Start\_Time = 00:00:00:00, Sector\_Offset = 0, Time\_Offset = 00:00:00:00)

VOBU\_Map\_Table = (Num\_of\_VOBU\_Map = 7)

VOBU\_Map#1 = (Start\_Sector = 0, Start\_Time = 00:00, Ref\_Pict\_EA = 13)  
 VOBU\_Map#2 = (Start\_Sector = 46, Start\_Time = 00:15, Ref\_Pict\_EA = 14)  
 VOBU\_Map#3 = (Start\_Sector = 0, Start\_Time = 00:00, Ref\_Pict\_EA = 11)  
 VOBU\_Map#4 = (Start\_Sector = 45, Start\_Time = 00:15, Ref\_Pict\_EA = 16)  
 VOBU\_Map#5 = (Start\_Sector = 0, Start\_Time = 00:00, Ref\_Pict\_EA = 14)  
 VOBU\_Map#6 = (Start\_Sector = 49, Start\_Time = 00:15, Ref\_Pict\_EA = 12)  
 VOBU\_Map#7 = (Start\_Sector = 0, Start\_Time = 00:00, Ref\_Pict\_EA = 12)

Time\_Map\_Table = (Num\_of\_Time\_Map = 3, Time\_Unit = 1, Time\_Base = 00:00, Index\_Offset = 0)

Time\_Map#1 = (VOBU\_Map\_Index = 3, Start\_Sector = 86, Start\_Time = 00:00:01:00)  
 Time\_Map#2 = (VOBU\_Map\_Index = 5, Start\_Sector = 184, Start\_Time = 00:00:02:00)  
 Time\_Map#3 = (VOBU\_Map\_Index = 7, Start\_Sector = 277, Start\_Time = 00:00:03:00)

【図61】

```

AV File Manager Information = (AV_File_ID = "System_2", Sector Size= 191, Playback_Time = 00:00:01:15)
VOB Table = (Num_of_VOB = 1)

VOB_Information#1 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = 0, Time_Offset = 00:00:00:00)
VOBU_Map_Table = (Num_of_VOBU_Map = 3)

VOBU_Map#1 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 13)
VOBU_Map#2 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 14)
VOBU_Map#3 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 11)

Time_Map_Table = (Num_of_Time_Map = 1, Time_Unit = 1, Time_Base = 00:00:00:00, Index_Offset = 0)
Time_Map#1 = (VOBU_Map_Index = 3, Start_Sector = 86, Start_Time = 00:00:01:00)

AV File Manager Information = (AV_File_ID = "System_2a", Sector Size= 191, Playback_Time = 00:00:02:00)
VOB Table = (Num_of_VOB = 1)

VOB_Information#1 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = -191, Time_Offset = -00:00:01:15)
VOBU_Map_Table = (Num_of_VOBU_Map = 4)

VOBU_Map#1 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 16)
VOBU_Map#2 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 14)
VOBU_Map#3 = (Start_Sector = 49, Start_Time = 00:15, Ref_Pict_EA = 12)
VOBU_Map#4 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 12)

Time_Map_Table = (Num_of_Time_Map = 2, Time_Unit = 1, Time_Base = -00:15, Index_Offset = -3)
Time_Map#1 = (VOBU_Map_Index = 5, Start_Sector = 184, Start_Time = 00:00:02:00)
Time_Map#2 = (VOBU_Map_Index = 7, Start_Sector = 277, Start_Time = 00:00:03:00)

```

【図62】

```

AV File Manager Information = (AV_File_ID = "System_3", Sector Size= 183, Playback_Time = 00:00:02:00)
VOB Table = (Num_of_VOB = 1)

VOB_Information#1 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = 0, Time_Offset = 00:00:00:00)
VOBU_Map_Table = (Num_of_VOBU_Map = 4)

VOBU_Map#1 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 12)
VOBU_Map#2 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 13)
VOBU_Map#3 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 12)
VOBU_Map#4 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 14)

Time_Map_Table = (Num_of_Time_Map = 1, Time_Unit = 1, Time_Base = 00:00, Index_Offset = 0)
Time_Map#1 = (VOBU_Map_Index = 3, Start_Sector = 87, Start_Time = 01:00)

AV File Manager Information = (AV_File_ID = "System_4", Sector Size= 133, Playback_Time = 00:00:01:15)
VOB Table = (Num_of_VOB = 1)

VOB_Information#1 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = 0, Time_Offset = 00:00:00:00)
VOBU_Map_Table = (Num_of_VOBU_Map = 3)

VOBU_Map#1 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 13)
VOBU_Map#2 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pict_EA = 14)
VOBU_Map#3 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pict_EA = 12)

Time_Map_Table = (Num_of_Time_Map = 2, Time_Unit = 1, Time_Base = 00:00, Index_Offset = 0)
Time_Map#1 = (VOBU_Map_Index = 3, Start_Sector = 88, Start_Time = 00:00:01:00)

```

【図65】

## 録画条件 AVデータ入力部における設定

高画質	ビットレート = 6Mbps・解像度 = 720 x 480
標準	ビットレート = 3Mbps・解像度 = 360 x 480
時間優先	ビットレート = 1.5Mbps・解像度 = 360 x 240

【図 6 3】

```

AV_Pt6 Manager Information = (AV_File_ID = "System_3", Sector_Size = 412, Playback_Time = 00:00:04:15)
VOB Table = (Num_of_VOB = 2)

VOB_Information01 = (Start_Sector = 0, Start_Time = 00:00:00:00, Sector_Offset = 0, Time_Offset = 00:00:00:00)
VOBU_Map_Table = (Num_of_VOBU_Map = 5)

VOBU_Map01 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pt6_EA = 12)
VOBU_Map02 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pt6_EA = 13)
VOBU_Map03 = (Start_Sector = 0, Start_Time = 00:15, Ref_Pt6_EA = 12)
VOBU_Map04 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pt6_EA = 14)
VOBU_Map05 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pt6_EA = 13)

Time_Map_Table = (Num_of_Time_Map = 2, Time_Unit = 1, Time_Base = 00:00, Index_Offset = 0)
Time_Map01 = (VOBU_Map_Index = 3, Start_Sector = 87, Start_Time = 00:00:01:00)
Time_Map02 = (VOBU_Map_Index = 5, Start_Sector = 183, Start_Time = 00:00:02:00)

VOB_Information02 = (Start_Sector = 223, Start_Time = 00:00:02:12, Sector_Offset = 56, Time_Offset = 00:00:00:15)
VOBU_Map_Table = (Num_of_VOBU_Map = 4)

VOBU_Map01 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pt6_EA = 12)
VOBU_Map02 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pt6_EA = 13)
VOBU_Map03 = (Start_Sector = 45, Start_Time = 00:15, Ref_Pt6_EA = 14)
VOBU_Map04 = (Start_Sector = 0, Start_Time = 00:00, Ref_Pt6_EA = 12)

Time_Map_Table = (Num_of_Time_Map = 2, Time_Unit = 1, Time_Base = -00:12, Index_Offset = 1)
Time_Map01 = (VOBU_Map_Index = 1, Start_Sector = 0, Start_Time = 00:00:00:00)
Time_Map02 = (VOBU_Map_Index = 3, Start_Sector = 88, Start_Time = 00:00:01:00)

```

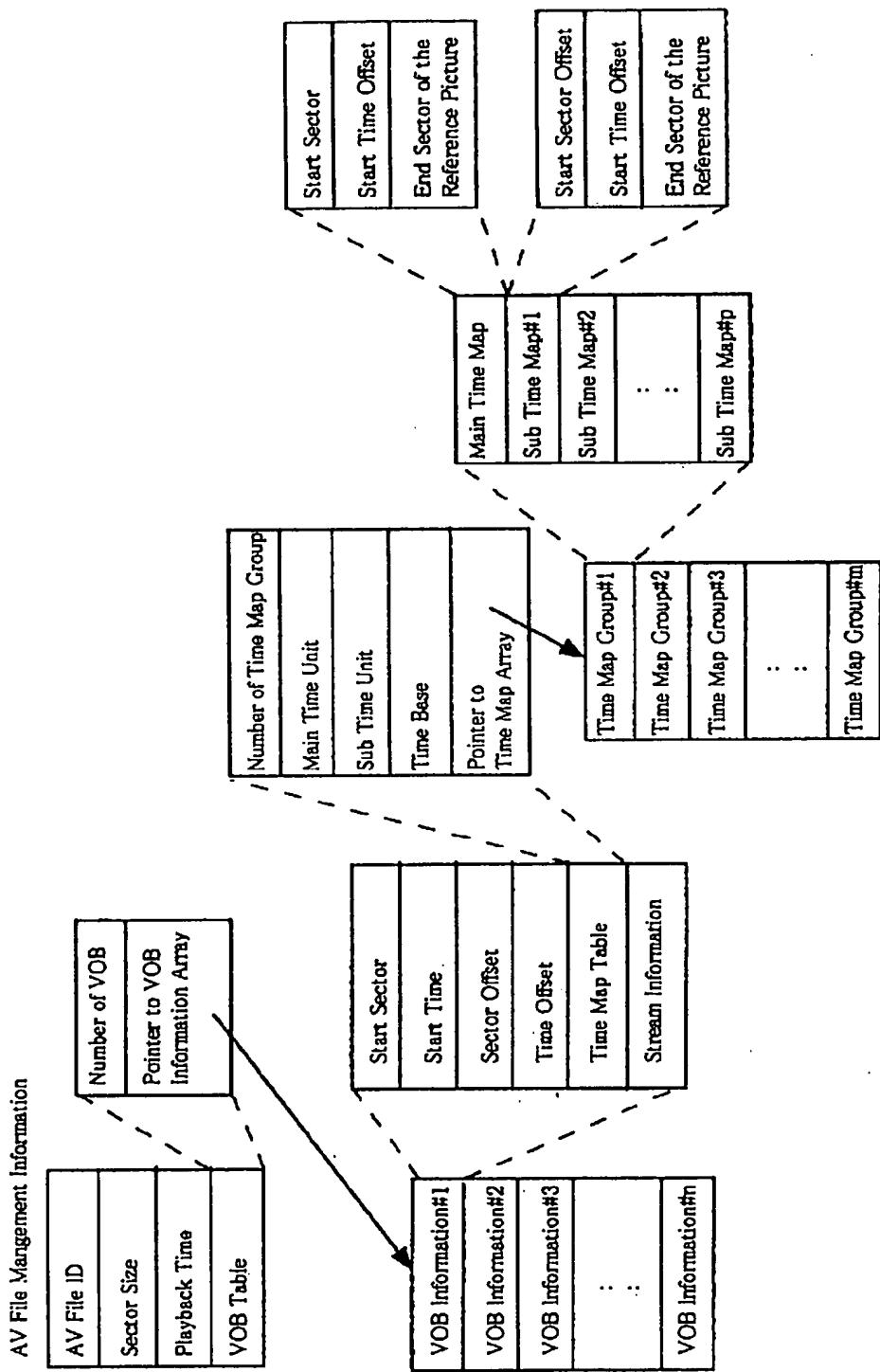
【図 6 7】

Stream Information	Stream Information
Video Attribute = (MPEG2, NTSC, 720x480, 4:3, ACC)	Video Attribute = (MPEG2, NTSC, 720x480, 4:3, ACC)
Audio Map Table	Audio Map Table
Audio Map01 = (Valid, 1)	Audio Map01 = (Valid, 1)
Audio Map02 = (Valid, 2)	Audio Map02 = (Valid, 2)
Audio Map03 = (Valid, 3)	Audio Map03 = (Invalid)
Audio Map04 = (Invalid)	Audio Map04 = (Invalid)
:	:
Audio Map08 = (Invalid)	Audio Map08 = (Invalid)
Audio Attribute Table	Audio Attribute Table
Audio Attribute01 = (Valid, MPEG1/L2, 48KHz, 192Kbps)	Audio Attribute01 = (Valid, AC3, 48KHz, 384Kbps)
Audio Attribute02 = (Valid, AC3, 48KHz, 384Kbps)	Audio Attribute02 = (Valid, MPEG1/L2, 48KHz, 192Kbps)
Audio Attribute03 = (Valid, AC3, 48KHz, 384Kbps)	Audio Attribute03 = (Invalid)
Audio Attribute04 = (Invalid)	Audio Attribute04 = (Invalid)
:	:
Audio Attribute08 = (Invalid)	Audio Attribute08 = (Invalid)

【図 6 8】

Stream Information	Stream Information
Video Attribute = (MPEG2, NTSC, 720x480, 4:3, ACC)	Video Attribute = (MPEG2, NTSC, 720x480, 4:3, ACC)
Audio Map Table	Audio Map Table
Audio Map01 = (Valid, 1)	Audio Map01 = (Valid, 2)
Audio Map02 = (Valid, 3)	Audio Map02 = (Valid, 1)
Audio Map03 = (Valid, 3)	Audio Map03 = (Valid, 1)
Audio Map04 = (Invalid)	Audio Map04 = (Invalid)
:	:
Audio Map08 = (Invalid)	Audio Map08 = (Invalid)
Audio Attribute Table	Audio Attribute Table
Audio Attribute01 = (Valid, MPEG1/L2, 48KHz, 192Kbps)	Audio Attribute01 = (Valid, AC3, 48KHz, 384Kbps)
Audio Attribute02 = (Valid, AC3, 48KHz, 384Kbps)	Audio Attribute02 = (Valid, MPEG1/L2, 48KHz, 192Kbps)
Audio Attribute03 = (Valid, AC3, 48KHz, 384Kbps)	Audio Attribute03 = (Invalid)
Audio Attribute04 = (Invalid)	Audio Attribute04 = (Invalid)
:	:
Audio Attribute08 = (Invalid)	Audio Attribute08 = (Invalid)

【図69】



フロントページの続き

(72) 発明者 中谷 徳夫

大阪府門真市大字門真1006番地 松下電器  
産業株式会社内